

日 本 国 特 許 庁  
JAPAN PATENT OFFICE



別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2000年 6月 8日

#4  
12-7-01

出 願 番 号

Application Number:

特願2000-172676

出 願 人

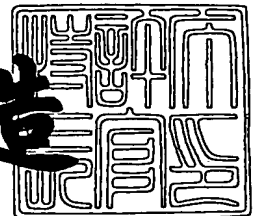
Applicant(s):

ソニー株式会社

2001年 5月11日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3040017

【書類名】 特許願

【整理番号】 0000118202

【提出日】 平成12年 6月 8日

【あて先】 特許庁長官 殿

【国際特許分類】 H03M 13/12

【発明者】

    【住所又は居所】 東京都品川区北品川6丁目7番35号      ソニー株式会  
社内

    【氏名】 宮内 俊之

【発明者】

    【住所又は居所】 東京都品川区北品川6丁目7番35号      ソニー株式会  
社内

    【氏名】 服部 雅之

【特許出願人】

    【識別番号】 000002185

    【氏名又は名称】 ソニー株式会社

    【代表者】 出井 伸之

【代理人】

    【識別番号】 100067736

    【弁理士】

    【氏名又は名称】 小池 晃

【選任した代理人】

    【識別番号】 100086335

    【弁理士】

    【氏名又は名称】 田村 榮一

【選任した代理人】

    【識別番号】 100096677

    【弁理士】

    【氏名又は名称】 伊賀 誠司

【手数料の表示】

【予納台帳番号】 019530

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9707387

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 復号装置及び復号方法

【特許請求の範囲】

【請求項 1】 各ステートに少なくとも 3 つ以上のパスが到達する符号化がなされて受信した軟入力とされる受信値に基づいて任意のステートを通過する確率を対数表記した対数尤度を求め、上記対数尤度を用いて復号を行う復号装置であって、

各ステートに到達した少なくとも 3 つ以上のパスのうち、尤度の高い少なくとも 2 つ以上のパスを求め、これらの少なくとも 2 つ以上のパスの中から、最も尤度の高いパスである最尤パスを選択するパス選択手段を備えること

を特徴とする復号装置。

【請求項 2】 上記パス選択手段は、各ステートに到達した少なくとも 3 つ以上の全てのパスの中から選択した 2 つのパスの組み合わせの全てについて尤度の大小を比較する比較手段を有すること

を特徴とする請求項 1 記載の復号装置。

【請求項 3】 上記最尤パスに対応するデータと 2 番目に尤度の高いパスである準最尤パスに対応するデータとの差分値の絶対値を求めて選択する絶対値選択手段を備えること

を特徴とする請求項 1 記載の復号装置。

【請求項 4】 上記絶対値選択手段は、各ステートに到達した少なくとも 3 つ以上の全てのパスの中から選択した 2 つのパスの組み合わせの全てについて、差分値の絶対値を算出する絶対値算出手段を有し、

上記パス選択手段により各ステートに到達した少なくとも 3 つ以上の全てのパスの中から選択した 2 つのパスの組み合わせの全てについて尤度の大小を比較した結果を示す比較結果情報に基づいて、上記絶対値算出手段により算出した絶対値の大小関係を判別すること

を特徴とする請求項 3 記載の復号装置。

【請求項 5】 上記対数尤度を与えるために追加され、変数に対する 1 次元の関数で表される補正項を線形近似により算出する線形近似手段を備え、

上記線形近似手段は、上記絶対値選択手段から供給された上記最尤パスに対応するデータと上記準最尤パスに対応するデータとの差分値の絶対値を上記変数とすること

を特徴とする請求項3記載の復号装置。

【請求項6】 上記線形近似手段は、少なくとも上記変数に乗算すべき上記関数の傾きを表す係数を2のべき乗を用いて表現して上記補正項を算出すること

を特徴とする請求項5記載の復号装置。

【請求項7】 上記線形近似手段は、上記関数の傾きを表す係数を表現するべき数に応じて、入力されたデータの下位ビットから切り捨てること

を特徴とする請求項6記載の復号装置。

【請求項8】 上記線形近似手段は、上記関数の傾きを表す係数を $-2^{-k}$ で表現したとき、入力されたデータの下位1ビット目から下位kビット目までを切り捨てること

を特徴とする請求項6記載の復号装置。

【請求項9】 上記線形近似手段は、上記関数の切片を表す係数を2のべき乗を用いて表現して上記補正項を算出すること

を特徴とする請求項6記載の復号装置。

【請求項10】 上記線形近似手段は、上記関数の切片を表す係数を $2^m - 1$ で表現して上記補正項を算出すること

を特徴とする請求項9記載の復号装置。

【請求項11】 上記線形近似手段は、上記関数の傾きを表す係数を $-2^{-k}$ で表現したとき、入力されたnビットのデータの下位1ビット目から下位kビット目までを切り捨て、下位k+1ビット目から下位m+kビット目までのmビットを反転すること

を特徴とする請求項10記載の復号装置。

【請求項12】 上記補正項は、正值であること

を特徴とする請求項6記載の復号装置。

【請求項13】 上記線形近似手段は、上記補正項が負値として算出された場合には、上記補正項を0とすること

を特徴とする請求項 12 記載の復号装置。

【請求項 14】 上記対数尤度は、上記確率を自然対数を用いて対数表記したものであること

を特徴とする請求項 1 記載の復号装置。

【請求項 15】 上記受信値毎に、符号の出力パターンと上記受信値により決定される第 1 の確率を対数表記した第 1 の対数尤度を算出する第 1 の確率算出手段と、

上記第 1 の対数尤度に基づいて、上記受信値毎に、符号化開始状態から時系列順に各状態に至る第 2 の確率を対数表記した第 2 の対数尤度を算出する第 2 の確率算出手段と、

上記第 1 の対数尤度に基づいて、上記受信値毎に、打ち切り状態から時系列の逆順に各状態に至る第 3 の確率を対数表記した第 3 の対数尤度を算出する第 3 の確率算出手段とを備え、

上記第 2 の確率算出手段及び上記第 3 の確率算出手段は、それぞれ、上記パス選択手段を有すること

を特徴とする請求項 1 記載の復号装置。

【請求項 16】 上記第 1 の対数尤度と、上記第 2 の対数尤度と、上記第 3 の対数尤度とを用いて、各時刻における軟出力を対数表記した対数軟出力を算出する軟出力算出手段を備えること

を特徴とする請求項 15 記載の復号装置。

【請求項 17】 上記対数軟出力は、上記軟出力を自然対数を用いて対数表記したものであること

を特徴とする請求項 16 記載の復号装置。

【請求項 18】 上記第 2 の確率算出手段及び上記第 3 の確率算出手段は、それぞれ、上記最尤パスに対応するデータと 2 番目に尤度の高いパスである準最尤パスに対応するデータとの差分値の絶対値を求めて選択する絶対値選択手段を有すること

を特徴とする請求項 15 記載の復号装置。

【請求項 19】 上記第 2 の確率算出手段及び上記第 3 の確率算出手段は、そ

れぞれ、上記対数尤度を与えるために追加され、変数に対する1次元の関数で表される補正項を線形近似により算出する線形近似手段を有すること

を特徴とする請求項15記載の復号装置。

【請求項20】 上記対数尤度は、上記確率の積演算を対数の和演算に置き換えた計算を行うとともに、上記確率の和演算を対数の最大値演算を行うことで求められること

を特徴とする請求項1記載の復号装置。

【請求項21】 Max-Log-BJRアルゴリズムに基づく最大事後確率復号を行うこと

を特徴とする請求項20記載の復号装置。

【請求項22】 上記対数尤度は、上記確率の積演算を対数の和演算に置き換えた計算を行うとともに、上記確率の和演算を対数の最大値演算と上記関数の演算とを行うことで求められること

を特徴とする請求項5記載の復号装置。

【請求項23】 Log-BJRアルゴリズムに基づく最大事後確率復号を行うこと

を特徴とする請求項22記載の復号装置。

【請求項24】 畳み込み符号の復号を行うこと

を特徴とする請求項1記載の復号装置。

【請求項25】 各ステートに少なくとも3つ以上のパスが到達する符号化がなされて受信した軟入力とされる受信値に基づいて任意のステートを通過する確率を対数表記した対数尤度を求め、上記対数尤度を用いて復号を行う復号方法であって、

各ステートに到達した少なくとも3つ以上のパスのうち、尤度の高い少なくとも2つ以上のパスを求め、これらの少なくとも2つ以上のパスの中から、最も尤度の高いパスである最尤パスを選択するパス選択工程を備えること

を特徴とする復号方法。

【請求項26】 上記パス選択工程では、各ステートに到達した少なくとも3つ以上の全てのパスの中から選択した2つのパスの組み合わせの全てについて尤

度の大小を比較すること

を特徴とする請求項 25 記載の復号方法。

【請求項 27】 上記最尤パスに対応するデータと 2 番目に尤度の高いパスである準最尤パスに対応するデータとの差分値の絶対値を求めて選択する絶対値選択工程を備えること

を特徴とする請求項 25 記載の復号方法。

【請求項 28】 上記絶対値選択工程では、各ステートに到達した少なくとも 3 つ以上の全てのパスの中から選択した 2 つのパスの組み合わせの全てについて、差分値の絶対値を算出し、

上記パス選択工程にて各ステートに到達した少なくとも 3 つ以上の全てのパスの中から選択した 2 つのパスの組み合わせの全てについて尤度の大小を比較した結果を示す比較結果情報に基づいて、算出した絶対値の大小関係を判別すること  
を特徴とする請求項 27 記載の復号方法。

【請求項 29】 上記対数尤度を与えるために追加され、変数に対する 1 次元の関数で表される補正項を線形近似により算出する線形近似工程を備え、

上記線形近似工程では、上記絶対値選択工程にて選択された上記最尤パスに対応するデータと上記準最尤パスに対応するデータとの差分値の絶対値を上記変数とすること

を特徴とする請求項 27 記載の復号方法。

【請求項 30】 上記線形近似工程では、少なくとも上記変数に乗算すべき上記関数の傾きを表す係数を 2 のべき乗を用いて表現して上記補正項を算出すること

を特徴とする請求項 29 記載の復号方法。

【請求項 31】 上記線形近似工程では、上記関数の傾きを表す係数を表現すべき数に応じて、入力されたデータの下位ビットから切り捨てること

を特徴とする請求項 30 記載の復号方法。

【請求項 32】 上記線形近似工程では、上記関数の傾きを表す係数を  $-2^{-k}$  で表現したとき、入力されたデータの下位 1 ビット目から下位 k ビット目までを切り捨てること



を特徴とする請求項30記載の復号方法。

【請求項33】 上記線形近似工程では、上記関数の切片を表す係数を2のべき乗を用いて表現して上記補正項を算出すること

を特徴とする請求項30記載の復号方法。

【請求項34】 上記線形近似工程では、上記関数の切片を表す係数を $2^m - 1$ で表現して上記補正項を算出すること

を特徴とする請求項33記載の復号方法。

【請求項35】 上記線形近似工程では、上記関数の傾きを表す係数を $-2^{-k}$ で表現したとき、入力されたnビットのデータの下位1ビット目から下位kビット目までを切り捨て、下位k+1ビット目から下位m+kビット目までのmビットを反転すること

を特徴とする請求項34記載の復号方法。

【請求項36】 上記補正項は、正值であること

を特徴とする請求項30記載の復号方法。

【請求項37】 上記線形近似工程では、上記補正項が負値として算出された場合には、上記補正項を0とすること

を特徴とする請求項36記載の復号方法。

【請求項38】 上記対数尤度は、上記確率を自然対数を用いて対数表記したものであること

を特徴とする請求項25記載の復号方法。

【請求項39】 上記受信値毎に、符号の出力パターンと上記受信値により決定される第1の確率を対数表記した第1の対数尤度を算出する第1の確率算出工程と、

上記第1の対数尤度に基づいて、上記受信値毎に、符号化開始状態から時系列順に各状態に至る第2の確率を対数表記した第2の対数尤度を算出する第2の確率算出工程と、

上記第1の対数尤度に基づいて、上記受信値毎に、打ち切り状態から時系列の逆順に各状態に至る第3の確率を対数表記した第3の対数尤度を算出する第3の確率算出工程とを備え、

上記第 2 の確率算出工程及び上記第 3 の確率算出工程は、それぞれ、上記パス選択工程を有すること

を特徴とする請求項 2 5 記載の復号方法。

【請求項 4 0】 上記第 1 の対数尤度と、上記第 2 の対数尤度と、上記第 3 の対数尤度とを用いて、各時刻における軟出力を対数表記した対数軟出力を算出する軟出力算出工程を備えること

を特徴とする請求項 3 9 記載の復号方法。

【請求項 4 1】 上記対数軟出力は、上記軟出力を自然対数を用いて対数表記したものであること

を特徴とする請求項 4 0 記載の復号方法。

【請求項 4 2】 上記第 2 の確率算出工程及び上記第 3 の確率算出工程は、それぞれ、上記最尤パスに対応するデータと 2 番目に尤度の高いパスである準最尤パスに対応するデータとの差分値の絶対値を求めて選択する絶対値選択工程を有すること

を特徴とする請求項 3 9 記載の復号方法。

【請求項 4 3】 上記第 2 の確率算出工程及び上記第 3 の確率算出工程は、それぞれ、上記対数尤度を与えるために追加され、変数に対する 1 次元の関数で表される補正項を線形近似により算出する線形近似工程を有すること

を特徴とする請求項 3 9 記載の復号方法。

【請求項 4 4】 上記対数尤度は、上記確率の積演算を対数の和演算に置き換えた計算を行うとともに、上記確率の和演算を対数の最大値演算を行うことで求めること

を特徴とする請求項 2 5 記載の復号方法。

【請求項 4 5】 Max-Log-BJR アルゴリズムに基づく最大事後確率復号を行うこと

を特徴とする請求項 4 4 記載の復号方法。

【請求項 4 6】 上記対数尤度は、上記確率の積演算を対数の和演算に置き換えた計算を行うとともに、上記確率の和演算を対数の最大値演算と上記関数の演算とを行うことで求めること

を特徴とする請求項 2 9 記載の復号方法。

【請求項 4 7】 L o g - B C J R アルゴリズムに基づく最大事後確率復号を行うこと

を特徴とする請求項 4 6 記載の復号方法。

【請求項 4 8】 畳み込み符号の復号を行うこと

を特徴とする請求項 2 5 記載の復号方法。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、軟出力復号を行う復号装置及び復号方法に関する。

【 0 0 0 2 】

【従来の技術】

近年、接続符号における内符号の復号出力や繰り返し復号法における各繰り返し復号動作の出力を軟出力とすることで、シンボル誤り率を小さくする研究がなされており、それに適した復号法に関する研究が盛んに行われている。例えば畳み込み符号等の所定の符号を復号した際のシンボル誤り率を最小にする方法としては、「Bahl, Cocke, Jelinek and Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", IEEE Trans. Inf. Theory, vol. IT-20, pp. 284-287, Mar. 1974」に記載されている B C J R アルゴリズムが知られている。この B C J R アルゴリズムにおいては、復号結果として各シンボルを出力するのではなく、各シンボルの尤度を出力する。このような出力は、軟出力 (soft-output) と呼ばれる。以下、この B C J R アルゴリズムの内容について説明する。なお、以下の説明では、図 2 3 に示すように、デジタル情報を図示しない送信装置が備える符号化装置 2 0 1 により畳み込み符号化し、その出力を雑音のある無記憶通信路 2 0 2 を介して図示しない受信装置に入力して、この受信装置が備える復号装置 2 0 3 により復号し、観測する場合を考える。

【 0 0 0 3 】

まず、符号化装置 2 0 1 が備えるシフトレジスタの内容を表す M 個のステート (遷移状態) を  $m (0, 1, \dots, M-1)$  で表し、時刻  $t$  のステートを  $S_t$

で表す。また、1タイムスロットにkビットの情報が入力されるものとする、時刻tにおける入力を $i_t = (i_{t1}, i_{t2}, \dots, i_{tk})$ で表し、入力系統を $I_1^T = (i_1, i_2, \dots, i_T)$ で表す。このとき、ステート $m'$ からステート $m$ への遷移がある場合には、その遷移に対応する情報ビットを $i(m', m) = (i_1(m', m), i_2(m', m), \dots, i_k(m', m))$ で表す。さらに、1タイムスロットにnビットの符号が出力されるものとする、時刻tにおける出力を $x_t = (x_{t1}, x_{t2}, \dots, x_{tn})$ で表し、出力系統を $X_1^T = (x_1, x_2, \dots, x_T)$ で表す。このとき、ステート $m'$ からステート $m$ への遷移がある場合には、その遷移に対応する符号ビットを $x(m', m) = (x_1(m', m), x_2(m', m), \dots, x_n(m', m))$ で表す。

【0004】

符号化装置201による畳み込み符号化は、ステート $S_0=0$ から始まり、 $X_1^T$ を出力して $S_T=0$ で終了するものとする。ここで、各ステート間の遷移確率 $P_t(m|m')$ を次式(1)により定義する。

【0005】

【数1】

$$P_i(m|m') = Pr\{S_i = m | S_{i-1} = m'\} \quad (1)$$

【0006】

なお、上式(1)における右辺に示す $Pr\{A|B\}$ は、Bが生じた条件の下でのAが生じる条件付き確率である。この遷移確率 $P_t(m|m')$ は、次式(2)に示すように、入力iでステート $m'$ からステート $m$ へと遷移するときに、時刻tでの入力 $i_t$ がiである確率 $Pr\{i_t = i\}$ と等しいものである。

【0007】

【数 2】

$$P_i(m|m') = Pr\{i_t = i\} \quad (2)$$

【0008】

雑音のある無記憶通信路 202 は、 $X_1^T$  を入力とし、 $Y_1^T$  を出力する。ここで、1 タイムスロットに  $n$  ビットの受信値が出力されるものとする、時刻  $t$  における出力を  $y_t = (y_{t1}, y_{t2}, \dots, y_{tn})$  で表し、 $Y_1^T = (y_1, y_2, \dots, y_T)$  で表す。雑音のある無記憶通信路 202 の遷移確率は、全ての  $t$  ( $1 \leq t \leq T$ ) について、次式 (3) に示すように、各シンボルの遷移確率  $Pr\{y_j | x_j\}$  を用いて定義することができる。

【0009】

【数 3】

$$Pr\{Y_1^T | X_1^T\} = \prod_{j=1}^T Pr\{y_j | x_j\} \quad (3)$$

【0010】

ここで、次式 (4) のように  $\lambda_{tj}$  を定義する。この次式 (4) に示す  $\lambda_{tj}$  は、 $Y_1^T$  を受信した際の時刻  $t$  での入力情報の尤度を表し、本来求めるべき軟出力である。

【0011】

【数4】

$$\lambda_{ij} = \frac{Pr \{i_{ij} = 1 | Y_1^T\}}{Pr \{i_{ij} = 0 | Y_1^T\}} \quad (4)$$

【0012】

BCJRアルゴリズムにおいては、次式(5)乃至次式(7)に示すような確率 $\alpha_t$ 、 $\beta_t$ 及び $\gamma_t$ を定義する。なお、 $Pr \{A; B\}$ は、AとBとがともに生じる確率を表すものとする。

【0013】

【数5】

$$\alpha_i(m) = Pr \{S_i = m; Y_1^t\} \quad (5)$$

【0014】

【数6】

$$\beta_i(m) = Pr \{Y_{i+1}^T | S_i = m\} \quad (6)$$

【0015】

【数 7】

$$\gamma_i(m', m) = Pr \{S_i = m; y_i | S_{i-1} = m'\} \quad (7)$$

【0016】

ここで、これらの確率  $\alpha_t$ 、 $\beta_t$  及び  $\gamma_t$  の内容について、符号化装置 201 における状態遷移図であるトレリスを図 24 を用いて説明する。同図において、 $\alpha_{t-1}$  は、符号化開始状態  $S_0 = 0$  から受信値をもとに時系列順に算出した時刻  $t-1$  における各状態の通過確率に対応する。また、 $\beta_t$  は、符号化終了状態  $S_T = 0$  から受信値をもとに時系列の逆順に算出した時刻  $t$  における各状態の通過確率に対応する。さらに、 $\gamma_t$  は、時刻  $t$  における受信値と入力確率とをもとに算出した時刻  $t$  に状態間を遷移する各枝の出力の受信確率に対応する。

【0017】

これらの確率  $\alpha_t$ 、 $\beta_t$  及び  $\gamma_t$  を用いると、軟出力  $\lambda_{t,j}$  は、次式 (8) のように表すことができる。

【0018】

【数 8】

$$\lambda_{t,j} = \frac{\sum_{\substack{m', m \\ i_j(m', m)=1}} \alpha_i(m') \gamma_i(m', m) \beta_i(m)}{\sum_{\substack{m', m \\ i_j(m', m)=0}} \alpha_i(m') \gamma_i(m', m) \beta_i(m)} \quad (8)$$

【0019】

ところで、 $t = 1, 2, \dots, T$  について、次式 (9) が成立する。

【0020】

【数9】

$$\alpha_i(m) = \sum_{m'=0}^{M-1} \alpha_{i-1}(m') \gamma_i(m', m) \quad (9)$$

ただし、 $\alpha_0(0) = 1, \alpha_0(m) = 0 (m \neq 0)$

【0021】

同様に、 $t = 1, 2, \dots, T$ について、次式(10)が成立する。

【0022】

【数10】

$$\beta_t(m) = \sum_{m'=0}^{M-1} \beta_{t-1}(m') \gamma_t(m, m') \quad (10)$$

ただし、 $\beta_T(0) = 1, \beta_T(m) = 0 (m \neq 0)$

【0023】

さらに、 $\gamma_t$ について、次式(11)が成立する。

【0024】



【数 11】

$$\gamma_t(m', m) = \begin{cases} P(m | m') \cdot Pr\{y_t | x(m', m)\} \\ \quad = Pr\{i_t = i(m', m)\} \cdot Pr\{y_t | x(m', m)\} \\ \quad : \text{入力 } i \text{ で } m' \text{ から } m \text{ へ遷移する場合} \\ 0 \quad : \text{入力 } i \text{ で } m' \text{ から } m \text{ へ遷移しない場合} \end{cases} \quad (11)$$

【0025】

したがって、復号装置203は、BCJRアルゴリズムを適用して軟出力復号を行う場合には、これらの関係に基づいて、図25に示す一連の工程を経ることにより軟出力 $\lambda_t$ を求める。

【0026】

まず、復号装置203は、同図に示すように、ステップS201において、 $y_t$ を受信する毎に、上式(9)及び上式(11)を用いて、確率 $\alpha_t(m)$ 及び $\gamma_t(m', m)$ を算出する。

【0027】

続いて、復号装置203は、ステップS202において、系列 $Y_1^T$ の全てを受信した後に、上式(10)を用いて、全ての時刻 $t$ における各ステート $m$ について、確率 $\beta_t(m)$ を算出する。

【0028】

そして、復号装置203は、ステップS203において、ステップS201及びステップS202において算出した確率 $\alpha_t$ 、 $\beta_t$ 及び $\gamma_t$ を上式(8)に代入し、各時刻 $t$ における軟出力 $\lambda_t$ を算出する。

【0029】

復号装置203は、このような一連の処理を経ることによって、BCJRアルゴリズムを適用した軟出力復号を行うことができる。

【0030】

ところで、このようなBCJRアルゴリズムにおいては、確率を直接値として保持して演算を行う必要があり、積演算を含むために演算量が大きいという問題があった。そこで、演算量を削減する手法として、「Robertson, Villebrun and Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the domain", IEEE Int. Conf. on Communications, pp. 1009-1013, June 1995」に記載されているMax-Log-MAPアルゴリズム及びLog-MAPアルゴリズム（以下、Max-Log-BCJRアルゴリズム及びLog-BCJRアルゴリズムと称する。）がある。

【0031】

まず、Max-Log-BCJRアルゴリズムについて説明する。Max-Log-BCJRアルゴリズムは、確率 $\alpha_t$ 、 $\beta_t$ 並びに $\gamma_t$ 、及び軟出力 $\lambda_t$ を自然対数を用いて対数表記し、次式(12)に示すように、確率の積演算を対数の和演算に置き換えるとともに、次式(13)に示すように、確率の和演算を対数の最大値演算で近似するものである。なお、次式(13)に示す $\max(x, y)$ は、 $x$ 、 $y$ のうち大きい値を有するものを選択する関数である。

【0032】

【数12】

$$\log(e^x \cdot e^y) = x + y \quad (12)$$

【0033】

【数 13】

$$\log(e^x + e^y) = \max(x, y) \quad (13)$$

【0034】

ここで、記載を簡略化するため、自然対数を  $I$  と略記し、 $\alpha_t, \beta_t, \gamma_t, \lambda_t$  の自然対数値を、それぞれ、次式 (14) に示すように、 $I\alpha_t, I\beta_t, I\gamma_t, I\lambda_t$  と表すものとする。

【0035】

【数 14】

$$\begin{cases} I\alpha_t(m) = \log(\alpha_t(m)) \\ I\beta_t(m) = \log(\beta_t(m)) \\ I\gamma_t(m) = \log(\gamma_t(m)) \\ I\lambda_t = \log \lambda_t \end{cases} \quad (14)$$

【0036】

Max-Log-BJR アルゴリズムにおいては、これらの対数尤度 (log likelihood)  $I\alpha_t, I\beta_t, I\gamma_t$  を、それぞれ、次式 (15) 乃至次式 (17) に示すように近似する。ここで、次式 (15) における右辺のステート  $m'$  に

おける最大値  $\max$  は、状態  $m$  への遷移が存在する状態  $m'$  の中で求めるものとし、次式 (16) における右辺の状態  $m'$  における最大値  $\max$  は、状態  $m$  からの遷移が存在する状態  $m'$  の中で求めるものとする。

【0037】

【数15】

$$I\alpha_t(m) = \max_{m'} \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) \right) \quad (15)$$

【0038】

【数16】

$$I\beta_t(m) = \max_{m'} \left( I\beta_{t+1}(m') + I\gamma_{t+1}(m, m') \right) \quad (16)$$

【0039】

【数17】

$$I\gamma_t(m', m) = \log \left( Pr \left\{ i_t = i(m', m) \right\} \right) + \log \left( Pr \left\{ y_t | x(m', m) \right\} \right) \quad (17)$$

【0040】

また、Max-Log-BCLR アルゴリズムにおいては、対数軟出力  $I\lambda_t$  についても同様に、次式 (18) に示すように近似する。ここで、次式 (18) における右辺第1項の最大値  $\max$  は、入力が“1”のときに状態  $m$  への遷

移が存在する状態  $m'$  の中で求め、第2項の最大値  $\max$  は、入力が“0”のときに状態  $m$  への遷移が存在する状態  $m'$  の中で求めるものとする。

【0041】

【数18】

$$I\lambda_{t,j} = \max_{\substack{m', m \\ i_{j(m', m)} = 1}} \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) + I\beta_t(m) \right) - \max_{\substack{m', m \\ i_{j(m', m)} = 0}} \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) + \beta_t(m) \right) \quad (18)$$

【0042】

したがって、復号装置203は、Max-Log-BCJRアルゴリズムを適用して軟出力復号を行う場合には、これらの関係に基づいて、図26に示す一連の工程を経ることにより軟出力  $\lambda_t$  を求める。

【0043】

まず、復号装置203は、同図に示すように、ステップS211において、 $y_t$ を受信する毎に、上式(15)及び上式(17)を用いて、対数尤度  $I\alpha_t(m)$  及び  $I\gamma_t(m', m)$  を算出する。

【0044】

続いて、復号装置203は、ステップS212において、系列  $Y_1^T$  の全てを受信した後に、上式(16)を用いて、全ての時刻  $t$  における各状態  $m$  について、対数尤度  $I\beta_t(m)$  を算出する。

【0045】

そして、復号装置203は、ステップS213において、ステップS211及びステップS212において算出した対数尤度  $I\alpha_t$ 、 $I\beta_t$  及び  $I\gamma_t$  を上式(18)に代入し、各時刻  $t$  における対数軟出力  $I\lambda_t$  を算出する。

【0046】

復号装置203は、このような一連の処理を経ることによって、Max-Log-BCJRアルゴリズムを適用した軟出力復号を行うことができる。

【0047】

このように、Max-Log-BCJR アルゴリズムは、積演算が含まれないことから、BCJR アルゴリズムと比較して、演算量を大幅に削減することができる。

【0048】

つぎに、Log-BCJR アルゴリズムについて説明する。Log-BCJR アルゴリズムは、Max-Log-BCJR アルゴリズムによる近似の精度をより向上させたものである。具体的には、Log-BCJR アルゴリズムは、上式(13)に示した確率の和演算を次式(19)に示すように補正項を追加することで変形し、和演算の正確な対数値を求めるものである。ここでは、このような補正を log-sum 補正と称するものとする。

【0049】

【数19】

$$\log(e^x + e^y) = \max(x, y) + \log(1 + e^{-|x-y|}) \quad (19)$$

【0050】

ここで、上式(19)における左辺に示す演算を log-sum 演算と称するものとし、この log-sum 演算の演算子を、「S. S. Pietrobon, "Implementation and performance of a turbo/MAP decoder", Int. J. Satellite Commun., vol. 16, pp. 23-46, Jan.-Feb. 1998」に記載されている記数法を踏襲し、次式(20)に示すように、便宜上“#”（ただし、同論文では、“E”。）と表すものとする。さらに、log-sum 演算の累積加算演算の演算子を、次式(21)に示すように、“#Σ”（ただし、同論文では、“E”。）と表すものとする。

【0051】

【数 20】

$$x \# y = \log(e^x + e^y) \quad (20)$$

【0052】

【数 21】

$$\# \sum_{i=0}^{M-1} x_i = \left( \left( \cdots \left( (x_0 \# x_1) \# x_2 \right) \cdots \right) \# x_{M-1} \right) \quad (21)$$

【0053】

これらの演算子を用いると、Log-BCJR アルゴリズムにおける対数尤度  $I \alpha_t$ 、 $I \beta_t$  及び対数軟出力  $I \lambda_t$  は、それぞれ、次式 (22) 乃至次式 (24) に示すように表すことができる。なお、対数尤度  $I \gamma_t$  は、上式 (17) で表されるため、ここでは、その記述を省略する。

【0054】

【数 22】

$$I \alpha_t(m) = \# \sum_{m'=0}^{M-1} \left( I \alpha_{t-1}(m') + I \gamma_t(m', m) \right) \quad (22)$$

【0055】

【数 23】

$$I\beta_t(m) = \# \sum_{m'=0}^{M-1} \left( I\beta_{t+1}(m') + I\gamma_{t+1}(m, m') \right) \quad (23)$$

【0056】

【数 24】

$$\begin{aligned} I\lambda_{t_j}(m) = & \# \sum_{\substack{m', m \\ i_j(m', m)=1}} \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) + I\beta_t(m) \right) \\ & - \# \sum_{\substack{m', m \\ i_j(m', m)=0}} \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) + I\beta_t(m) \right) \end{aligned} \quad (24)$$

【0057】

なお、上式(22)における右辺のステート $m'$ におけるlog-sum演算の累積加算演算は、ステート $m$ への遷移が存在するステート $m'$ の中で求めるものとし、上式(23)における右辺のステート $m'$ におけるlog-sum演算の累積加算演算は、ステート $m$ からの遷移が存在するステート $m'$ の中で求めるものとする。また、上式(24)における右辺第1項のlog-sum演算の累積加算演算は、入力が“1”のときにステート $m$ への遷移が存在するステート $m'$ の中で求め、第2項のlog-sum演算の累積加算演算は、入力が“0”のときにステート $m$ への遷移が存在するステート $m'$ の中で求めるものとする。

【0058】

したがって、復号装置203は、Log-BCKRアルゴリズムを適用して軟出力復号を行う場合には、これらの関係に基づいて、先に図26に示した一連の工程を経ることにより軟出力 $\lambda_t$ を求めることができる。

【0059】

まず、復号装置203は、同図に示すように、ステップS211において、 $y$



$t$ を受信する毎に、上式(22)及び上式(17)を用いて、対数尤度 $I_{\alpha_t}(m)$ 及び $I_{\gamma_t}(m', m)$ を算出する。

【0060】

続いて、復号装置203は、ステップS212において、系列 $Y_1^T$ の全てを受信した後に、上式(23)を用いて、全ての時刻 $t$ における各状態 $m$ について、対数尤度 $I_{\beta_t}(m)$ を算出する。

【0061】

そして、復号装置203は、ステップS213において、ステップS211及びステップS212において算出した対数尤度 $I_{\alpha_t}$ 、 $I_{\beta_t}$ 及び $I_{\gamma_t}$ を上式(24)に代入し、各時刻 $t$ における対数軟出力 $I_{\lambda_t}$ を算出する。

【0062】

復号装置203は、このような一連の処理を経ることによって、Log-BCJRアルゴリズムを適用した軟出力復号を行うことができる。なお、上式(19)において、右辺第2項に示す補正項は、変数 $|x-y|$ に対する1次元の関数で表されることから、復号装置203は、この値を図示しないROM(Read Only Memory)等にテーブルとして予め記憶しておくことによって、正確な確率計算を行うことができる。

【0063】

このようなLog-BCJRアルゴリズムは、Max-Log-BCJRアルゴリズムと比較すると演算量は増えるものの積演算を含むものではなく、その出力は、量子化誤差を除けば、BCJRアルゴリズムの軟出力の対数値そのものに他ならない。

【0064】

【発明が解決しようとする課題】

ところで、上述したlog-sum補正の方法としては、上述したように、補正項の値をテーブル化しておく方法の他に、変数 $|x-y|$ との関係をいわゆる2次近似により近似する2次近似法、変数 $|x-y|$ を任意の区間に分割して各区間毎に所定の値を与える区間分割法等がある。補正項の値をテーブル化しておく方法を含め、これらの方法は、補正項の値をいかに正確に求めるかといった性

能を重視した補正である。しかしながら、これらの方法は、回路規模の増大や速度の遅延を招くといった問題があった。

## 【0065】

そこで、 $\log\text{-sum}$ 補正の方法としては、速度を重視した方法が検討されている。この方法としては、変数 $|x-y|$ との関係をいわゆる線形近似により近似する線形近似法、変数 $|x-y|$ における所定の区間の値を所定の閾値で決定する閾値近似法がある。

## 【0066】

線形近似法は、図27(A)に示すように、曲線Cに示す関数 $F = \log\{1 + e^{-(|x-y|)}\}$ を直線Lに示す線形関数で近似するものである。同図においては、直線Lとして、 $F = -0.3(|x-y|) + \log 2$ を用いており、この場合、約0.1dB程度の劣化で補正項を求めることができる。

## 【0067】

また、閾値近似法は、同図(B)に示すように、曲線Cに示す関数 $F = \log\{1 + e^{-(|x-y|)}\}$ を曲線Tに示す階段関数で近似するものである。同図においては、曲線Tとして、 $0 \leq |x-y| < 1$ の区間では $\log 2$ を与え、 $|x-y| \geq 1$ の区間では0を与えるような関数を用いている。この場合、約0.2dB程度の劣化で補正項を求めることができる。

## 【0068】

このように、 $\log\text{-sum}$ 補正としては、様々な方法が検討されているが、一方で、未だ改善の余地が残るのが現状である。

## 【0069】

また、 $\log\text{-sum}$ 補正を行う際に限らず、上述した最大値演算を行うことによって、最も尤度の高いパスである最尤パスを求める際にも遅延が生じ、処理の高速化を妨げる要因となっている。

## 【0070】

本発明は、このような実情に鑑みてなされたものであり、性能を劣化させることなく、高速化を図ることができる復号装置及び復号方法を提供することを目的とする。

## 【 0 0 7 1 】

## 【課題を解決するための手段】

上述した目的を達成する本発明にかかる復号装置は、各ステートに少なくとも3つ以上のパスが到達する符号化がなされて受信した軟入力とされる受信値に基づいて任意のステートを通過する確率を対数表記した対数尤度を求め、この対数尤度を用いて復号を行う復号装置であって、各ステートに到達した少なくとも3つ以上のパスのうち、尤度の高い少なくとも2つ以上のパスを求め、これらの少なくとも2つ以上のパスの中から、最も尤度の高いパスである最尤パスを選択するパス選択手段を備えることを特徴としている。

## 【 0 0 7 2 】

このような本発明にかかる復号装置は、パス選択手段によって、尤度の高い少なくとも2つ以上のパスを求め、最尤パスを選択する。

## 【 0 0 7 3 】

また、上述した目的を達成する本発明にかかる復号方法は、各ステートに少なくとも3つ以上のパスが到達する符号化がなされて受信した軟入力とされる受信値に基づいて任意のステートを通過する確率を対数表記した対数尤度を求め、この対数尤度を用いて復号を行う復号方法であって、各ステートに到達した少なくとも3つ以上のパスのうち、尤度の高い少なくとも2つ以上のパスを求め、これらの少なくとも2つ以上のパスの中から、最も尤度の高いパスである最尤パスを選択するパス選択工程を備えることを特徴としている。

## 【 0 0 7 4 】

このような本発明にかかる復号方法は、パス選択工程にて、尤度の高い少なくとも2つ以上のパスを求め、最尤パスを選択する。

## 【 0 0 7 5 】

## 【発明の実施の形態】

以下、本発明を適用した具体的な実施の形態について図面を参照しながら詳細に説明する。

## 【 0 0 7 6 】

この実施の形態は、図1に示すように、ディジタル情報を図示しない送信装置

が備える符号化装置1により畳み込み符号化し、その出力を雑音のある無記憶通信路2を介して図示しない受信装置に入力して、この受信装置が備える復号装置3により復号する通信モデルに適用したデータ送受信システムである。

# 【0077】

このデータ送受信システムにおいて、復号装置3は、符号化装置1により畳み込み符号化がなされた符号の復号を行うものであって、「Robertson, Villebrun and Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the domain", IEEE Int. Conf. on Communications, pp. 1009-1013, June 1995」に記載されているMax-Log-MAPアルゴリズム又はLog-MAPアルゴリズム（以下、Max-Log-BJRアルゴリズム又はLog-BJRアルゴリズムと称する。）に基づく最大事後確率（Maximum A Posteriori probability; 以下、MAPと記す。）復号を行うものとして構成され、いわゆる確率 $\alpha_t$ 、 $\beta_t$ 、 $\gamma_t$ 、及び軟出力（soft-output） $\lambda_t$ を自然対数を用いて対数表記した対数尤度（log likelihood） $I \alpha_t$ 、 $I \beta_t$ 、 $I \gamma_t$ 、及び対数軟出力 $I \lambda_t$ を求めるものである。

# 【0078】

なお、以下では、復号装置3は、Log-BJRアルゴリズムに基づくMAP復号を行い、いわゆる線形近似によるlog-sum補正を行うことで対数軟出力 $I \lambda_t$ を求めるものとして説明する。また、以下では、符号化装置1が備えるシフトレジスタの内容を表すM個のステート（遷移状態）を $m(0, 1, \dots, M-1)$ で表し、時刻tのステートを $S_t$ で表す。さらに、1タイムスロットにkビットの情報が入力されるものとする、時刻tにおける入力を $i_t = (i_{t1}, i_{t2}, \dots, i_{tk})$ で表し、入力系統を $I_1^T = (i_1, i_2, \dots, i_T)$ で表す。このとき、ステート $m'$ からステート $m$ への遷移がある場合には、その遷移に対応する情報ビットを $i(m', m) = (i_1(m', m), i_2(m', m), \dots, i_k(m', m))$ で表す。さらにまた、1タイムスロットにnビットの符号が出力されるものとする、時刻tにおける出力を $x_t = (x_{t1}, x_{t2}, \dots, x_{tn})$ で表し、出力系統を $X_1^T = (x_1, x_2, \dots, x_T)$ で表す。このとき、ステート $m'$ からステート $m$ への遷移がある場合には、その

遷移に対応する符号ビットを  $x(m', m) = (x_1(m', m), x_2(m', m), \dots, x_n(m', m))$  で表す。また、無記憶通信路 2 は、 $X_1^T$  を入力とし、 $Y_1^T$  を出力するものとする。ここで、1 タイムスロットに  $n$  ビットの受信値が出力されるものとする、時刻  $t$  における出力を  $y_t = (y_{t1}, y_{t2}, \dots, y_{tn})$  で表し、 $Y_1^T = (y_1, y_2, \dots, y_T)$  で表す。

【0079】

符号化装置 1 は、例えば図 2 に示すように、3 つの排他的論理和回路 11, 13, 15 と、2 つのシフトレジスタ 12, 14 とを有し、拘束長が“4”の畳み込み演算を行うものとして構成される。

【0080】

排他的論理和回路 11 は、2 ビットの入力データ  $i_t$  のうちの 1 ビットの入力データ  $i_{t1}$  と、排他的論理和回路 15 から供給されるデータとを用いて排他的論理和演算を行い、演算結果をシフトレジスタ 12 に供給する。

【0081】

シフトレジスタ 12 は、保持している 1 ビットのデータを排他的論理和回路 13 に供給し続ける。そして、シフトレジスタ 12 は、クロックに同期させて、排他的論理和回路 11 から供給される 1 ビットのデータを新たに保持し、このデータを排他的論理和回路 13 に新たに供給する。

【0082】

排他的論理和回路 13 は、2 ビットの入力データ  $i_t$  のうちの 1 ビットの入力データ  $i_{t2}$  と、シフトレジスタ 12 から供給されるデータと、排他的論理和回路 15 から供給されるデータとを用いて排他的論理和演算を行い、演算結果をシフトレジスタ 14 に供給する。

【0083】

シフトレジスタ 14 は、保持している 1 ビットのデータを排他的論理和回路 15 に供給し続ける。そして、シフトレジスタ 14 は、クロックに同期させて、排他的論理和回路 13 から供給される 1 ビットのデータを新たに保持し、このデータを排他的論理和回路 15 に新たに供給する。

【0084】

排他的論理和回路 15 は、2 ビットの入力データ  $i_{t1}$ ,  $i_{t2}$  と、シフトレジスタ 14 から供給されるデータとを用いて排他的論理和演算を行い、演算結果を 3 ビットの出力データ  $x_t$  のうちの 1 ビットの出力データ  $x_{t3}$  として外部に出力するとともに、排他的論理和回路 11, 13 に供給する。

## 【0085】

このような符号化装置 1 は、2 ビットの入力データ  $i_{t1}$ ,  $i_{t2}$  を入力すると、これらの入力データ  $i_{t1}$ ,  $i_{t2}$  を、それぞれ、3 ビットの出力データ  $x_t$  のうちの組織成分の 2 ビットの出力データ  $x_{t1}$ ,  $x_{t2}$  として、そのまま外部に出力するとともに、入力データ  $i_{t1}$ ,  $i_{t2}$  に対して再帰的畳み込み演算を行い、演算結果を 3 ビットの出力データ  $x_t$  のうちの 1 ビットの出力データ  $x_{t3}$  として外部に出力する。すなわち、符号化装置 1 は、符号化率が“2/3”の再帰的組織畳み込み演算を行い、出力データ  $x_t$  を外部に出力する。

## 【0086】

この符号化装置 1 におけるトレリスを記述すると、図 3 に示すようになる。同図において、各パスに付与されているラベルは、2 ビットの入力データ  $i_{t1}$ ,  $i_{t2}$  / 3 ビットの出力データ  $x_{t1}$ ,  $x_{t2}$ ,  $x_{t3}$  を示している。ここでは、ステートは、シフトレジスタ 12 の内容とシフトレジスタ 14 の内容とを順次並べたものであり、“00”、“10”、“01”、“11”のステート番号を、それぞれ、“0”、“1”、“2”、“3”と表している。このように、符号化装置 1 におけるステート数  $M$  は 4 となり、トレリスは、各ステートから次時刻におけるステートへと 4 本のパスが到達する構造を有する。なお、以下の説明では、各ステート番号に対応するステートを指示する場合には、それぞれ、ステート 0、ステート 1、ステート 2、ステート 3 と称するものとする。

## 【0087】

このような符号化装置 1 により符号化された出力データ  $x_t$  は、無記憶通信路 2 を介して受信装置に出力される。

## 【0088】

一方、復号装置 3 は、図 4 に示すように、各部を制御するコントローラ 31 と、第 1 の対数尤度である対数尤度  $I_r$  を算出して記憶する第 1 の確率算出手段で

ある  $I_\gamma$  算出・記憶回路 32 と、第 2 の対数尤度である対数尤度  $I_\alpha$  を算出して記憶する第 2 の確率算出手段である  $I_\alpha$  算出・記憶回路 33 と、第 3 の対数尤度である対数尤度  $I_\beta$  を算出して記憶する第 3 の確率算出手段である  $I_\beta$  算出・記憶回路 34 と、対数軟出力  $I_{\lambda_t}$  を算出する軟出力算出手段である軟出力算出回路 35 とを備える。この復号装置 3 は、無記憶通信路 2 上で発生したノイズの影響によりアナログ値をとり軟入力 (soft-input) とされる受信値  $y_t$  から対数軟出力  $I_{\lambda_t}$  を求めることによって、符号化装置 1 における入力データ  $i_t$  を推定するものである。

【0089】

コントローラ 31 は、 $I_\gamma$  算出・記憶回路 32、 $I_\alpha$  算出・記憶回路 33 及び  $I_\beta$  算出・記憶回路 34 に対して、それぞれ、コントロール信号  $SC_\gamma$ 、 $SC_\alpha$  及び  $SC_\beta$  を供給し、各部の動作を制御する。

【0090】

$I_\gamma$  算出・記憶回路 32 は、コントローラ 31 から供給されたコントロール信号  $SC_\gamma$  による制御の下に、受信値  $y_t$  と、事前確率情報 (a priori probability information)  $Pr_t$  とを用いて、受信値  $y_t$  毎に、次式 (25) に示す演算を行い、各時刻  $t$  における対数尤度  $I_{\gamma_t}$  を算出して記憶する。すなわち、 $I_\gamma$  算出・記憶回路 32 は、受信値  $y_t$  毎に、符号の出力パターンと受信値により決定される確率  $\gamma$  を対数表記した対数尤度  $I_\gamma$  を算出する。

【0091】

【数 25】

$$I_{\gamma_t}(m', m) = \log \left( Pr \left\{ i_t = i(m', m) \right\} \right) + \log \left( Pr \left\{ y_t | x(m', m) \right\} \right) \quad (25)$$

【0092】

なお、事前確率情報  $Pr_t$  は、次式 (26) に示すように、入力データ  $i_{t1}$ ,

$i_{t2}$ のそれぞれが“1”である確率 $Pr\{i_t=1\}$ 又は入力データ $i_{t1}$ ,  $i_{t2}$ のそれぞれが“0”である確率 $Pr\{i_t=0\}$ として与えられる。また、事前確率情報 $Pr_t$ は、確率 $Pr\{i_t=1\}$ と確率 $Pr\{i_t=0\}$ との比の自然対数値である対数尤度比(log likelihood ratio)を入力し、確率 $Pr\{i_t=1\}$ と確率 $Pr\{i_t=0\}$ との和が“1”であることを考慮して、確率 $Pr\{i_t=1\}$ 又は確率 $Pr\{i_t=0\}$ として求められてもよい。

【0093】

【数26】

$$Pr_t = \begin{cases} \log Pr\{i_t=1\} \\ \log Pr\{i_t=0\} \end{cases} \quad (26)$$

【0094】

そして、 $I_\gamma$ 算出・記憶回路32は、記憶した対数尤度 $I_{\gamma t}$ を $I_\alpha$ 算出・記憶回路33、 $I_\beta$ 算出・記憶回路34及び軟出力算出回路35に供給する。このとき、 $I_\gamma$ 算出・記憶回路32は、 $I_\alpha$ 算出・記憶回路33、 $I_\beta$ 算出・記憶回路34及び軟出力算出回路35のそれぞれにおける処理に適した順序で対数尤度 $I_{\gamma t}$ を供給する。なお、以下の説明では、 $I_\gamma$ 算出・記憶回路32から $I_\alpha$ 算出・記憶回路33に供給される対数尤度 $I_{\gamma t}$ を $I_\gamma(\alpha)$ と表し、 $I_\gamma$ 算出・記憶回路32から $I_\beta$ 算出・記憶回路34に供給される対数尤度 $I_{\gamma t}$ を $I_\gamma(\beta 1)$ ,  $I_\gamma(\beta 2)$ と表し、 $I_\gamma$ 算出・記憶回路32から軟出力算出回路35に供給される対数尤度 $I_{\gamma t}$ を $I_\gamma(\lambda)$ と表すものとする。

【0095】

$I_\alpha$ 算出・記憶回路33は、コントローラ31から供給されたコントロール信号 $SC_\alpha$ による制御の下に、 $I_\gamma$ 算出・記憶回路32から供給された対数尤度 $I_\gamma(\alpha)$ を用いて、次式(27)に示す演算を行い、各時刻 $t$ における対数尤度 $I_{\alpha t}$ を算出して記憶する。なお、次式(27)における演算子“#”は、いわゆるlog-sum演算を示すものであり、入力“0”でステート $m'$ からステ



ート $m$ へと遷移するときにおける対数尤度と、入力“1”でステート $m'$ からステート $m$ へと遷移するときにおける対数尤度との $\log$ - $\text{sum}$ 演算を示すものである。より具体的には、 $I\alpha$ 算出・記憶回路33は、次式(28)に示す演算を行うことによって、各時刻 $t$ における対数尤度 $I\alpha_t$ を算出する。すなわち、 $I\alpha$ 算出・記憶回路33は、対数尤度 $I\gamma$ に基づいて、受信値 $y_t$ 毎に、符号化開始ステートから時系列順に各ステートに至る確率 $\alpha$ を対数表記した対数尤度 $I\alpha$ を算出する。そして、 $I\alpha$ 算出・記憶回路33は、記憶した対数尤度 $I\alpha_t$ を軟出力算出回路35に供給する。このとき、 $I\alpha$ 算出・記憶回路33は、軟出力算出回路35における処理に適した順序で対数尤度 $I\alpha_t$ を供給する。なお、以下の説明では、 $I\alpha$ 算出・記憶回路33から軟出力算出回路35に供給される対数尤度 $I\alpha_t$ を $I\alpha(\lambda)$ と表すものとする。

【0096】

【数27】

$$I\alpha_t(m) = \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) \right) \# \left( I\alpha_{t-1}(m'') + I\gamma_t(m'', m) \right) \quad (27)$$

【0097】

【数28】

$$I\alpha_t(m) = \max \left( I\alpha_{t-1}(m') + I\gamma_t(m', m), I\alpha_{t-1}(m'') + I\gamma_t(m'', m) \right) + \log \left( \frac{e^{-\left| \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) \right) - \left( I\alpha_{t-1}(m'') + I\gamma_t(m'', m) \right) \right|}}{1 + e} \right) \quad (28)$$

【0098】

$I\beta$ 算出・記憶回路34は、コントローラ31から供給されたコントロール信号 $SC\beta$ による制御の下に、 $I\gamma$ 算出・記憶回路32から供給された対数尤度 $I$

$\gamma(\beta 1)$ ,  $I\gamma(\beta 2)$  を用いて、次式(29)に示す演算を行い、各時刻における2系統の対数尤度  $I\beta_t$  を並列的に算出して記憶する。なお、次式(29)における演算子“#”は、上述したように、 $\log\text{-sum}$  演算を示すものであり、入力“0”でステート  $m'$  からステート  $m$  へと遷移するときにおける対数尤度と、入力“1”でステート  $m''$  からステート  $m$  へと遷移するときにおける対数尤度との  $\log\text{-sum}$  演算を示すものである。より具体的には、 $I\beta$  算出・記憶回路34は、次式(30)に示す演算を行うことによって、各時刻  $t$  における対数尤度  $I\beta_t$  を算出する。すなわち、 $I\beta$  算出・記憶回路34は、対数尤度  $I\gamma$  に基づいて、受信値  $y_t$  毎に、打ち切りステートから時系列の逆順に各ステートに至る確率  $\beta$  を対数表記した対数尤度  $I\beta$  を算出する。そして、 $I\beta$  算出・記憶回路34は、記憶した対数尤度  $I\beta_t$  のうち、1系統の対数尤度  $I\beta_t$  を軟出力算出回路35に供給する。このとき、 $I\beta$  算出・記憶回路34は、軟出力算出回路35における処理に適した順序で対数尤度  $I\beta_t$  を供給する。なお、以下の説明では、 $I\beta$  算出・記憶回路34から軟出力算出回路35に供給される対数尤度  $I\beta_t$  を  $I\beta(\lambda)$  と表すものとする。

【0099】

【数29】

$$I\beta_t(m) = \left( I\beta_{t+1}(m') + I\gamma_{t+1}(m, m') \right) \# \left( I\beta_{t+1}(m'') + I\gamma_{t+1}(m, m'') \right) \quad (29)$$

【0100】

【数 30】

$$I\beta_t(m) = \max \left( I\beta_{t+1}(m') + I\gamma_{t+1}(m, m'), I\beta_{t+1}(m'') + I\gamma_{t+1}(m, m'') \right) + \log \left( \frac{1}{1 + e^{-\left| \left( I\beta_{t+1}(m') + I\gamma_{t+1}(m, m') \right) - \left( I\beta_{t+1}(m'') + I\gamma_{t+1}(m, m'') \right) \right|}} \right) \quad (30)$$

【0101】

軟出力算出回路35は、 $I\gamma$ 算出・記憶回路32から供給された対数尤度 $I\gamma(\lambda)$ と、 $I\alpha$ 算出・記憶回路33から供給された対数尤度 $I\alpha(\lambda)$ と、 $I\beta$ 算出・記憶回路34から供給された対数尤度 $I\beta(\lambda)$ とを用いて、次式(31)に示す演算を行い、各時刻における対数軟出力 $I\lambda_t$ を算出して記憶する。そして、軟出力算出回路35は、記憶した対数軟出力 $I\lambda_t$ を時系列順に並べ替えた後、外部に出力する。なお、次式(31)における演算子“ $\# \Sigma$ ”は、上述した演算子“ $\#$ ”で表される $\log - \text{sum}$ 演算の累積加算演算を示すものである。

【0102】

【数 31】

$$I\lambda_t = \# \sum_{\substack{m', m \\ i(m', m)=1}} \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) + I\beta_t(m) \right) - \# \sum_{\substack{m', m \\ i(m', m)=0}} \left( I\alpha_{t-1}(m') + I\gamma_t(m', m) + I\beta_t(m) \right) \quad (31)$$

【0103】

このような復号装置3は、受信装置により受信された軟入力を受信値 $y_t$ を入力すると、 $I\gamma$ 算出・記憶回路32によって、受信値 $y_t$ を受信する毎に、対数尤度 $I\gamma_t(m', m)$ を算出し、 $I\alpha$ 算出・記憶回路33によって、対数尤度 $I\alpha_t(m)$ を算出した後、全ての受信値 $y_t$ を受信すると、 $I\beta$ 算出・記憶回路

34によって、全ての時刻 $t$ における各ステート $m$ について、対数尤度 $I\beta_t(m)$ を算出する。そして、復号装置3は、軟出力算出回路35によって、算出した対数尤度 $I\alpha_t$ 、 $I\beta_t$ 及び $I\gamma_t$ を用いて、各時刻 $t$ における対数軟出力 $I\lambda_t$ を算出する。このように、復号装置3は、Log-BCJRアルゴリズムを適用した軟出力復号を行うことができる。

#### 【0104】

さて、復号装置3は、 $I\alpha$ 算出・記憶回路33及び $I\beta$ 算出・記憶回路34における対数尤度 $I\alpha_t$ 、 $I\beta_t$ の算出の際の高速化を図ることができるものである。回路規模を削減したものである。そこで、以下では、 $I\alpha$ 算出・記憶回路33及び $I\beta$ 算出・記憶回路34について、より具体的に詳述する。

#### 【0105】

まず、 $I\alpha$ 算出・記憶回路33について説明する。 $I\alpha$ 算出・記憶回路33は、図5に示すように、算出された対数尤度 $I\alpha$ と対数尤度の初期値 $I\alpha_0$ とのいずれか一方を選択するセレクタ41と、初期値 $I\alpha_0$ 又は対数尤度 $I\alpha$ を保持するレジスタ42と、各ステートにおける対数尤度 $I\alpha$ を算出する $I\alpha$ 算出回路43と、各ステートにおける対数尤度 $I\alpha$ を順次保持するRAM(Random Access Memory)44、45と、これらのRAM44、45から読み出した対数尤度 $I\alpha$ を選択的に取り出す選択回路46とを有する。

#### 【0106】

セレクタ41は、コントローラ31から供給されたコントロール信号 $SC\alpha$ による制御の下に、初期化時には対数尤度の初期値 $I\alpha_0$ を選択し、初期化時以外の時には $I\alpha$ 算出回路43から供給される対数尤度 $I\alpha$ を選択する。なお、初期化は、 $I\gamma$ 算出・記憶回路32からの対数尤度 $I\gamma(\alpha)$ の出力が開始される1時刻前の時点で行われる。ここで、復号装置3が符号化装置1による符号化の開始時点を把握している場合には、初期値 $I\alpha_0$ としては、ステート0における値として $\log 1 = 0$ が与えられ、その他のステートにおける値として $\log 0 = -\infty$ が与えられる。一方、復号装置3が符号化装置1による符号化の開始時点を把握していない場合には、初期値 $I\alpha_0$ としては、全てのステートに対して $\log(1/M)$ 、ここでは $\log(1/4)$ が与えられるが、実際には、全てのス

テートに対して同じ値が与えられればよく、例えば全てのステートに対して0が与えられてもよい。セクタ41は、初期値 $I\alpha_0$ 又は対数尤度 $I\alpha$ のうちの選択した一方をレジスタ42に供給する。

## 【0107】

レジスタ42は、セクタ41から供給される初期値 $I\alpha_0$ 又は対数尤度 $I\alpha$ を保持する。そして、レジスタ42は、次時刻において、保持している初期値 $I\alpha_0$ 又は対数尤度 $I\alpha$ を $I\alpha$ 算出回路43及びRAM44, 45に供給する。

## 【0108】

$I\alpha$ 算出回路43は、図6に示すように、各ステートに応じた数、ここでは4つの加算比較選択回路47<sub>0</sub>, 47<sub>1</sub>, 47<sub>2</sub>, 47<sub>3</sub>を有する。

## 【0109】

加算比較選択回路47<sub>0</sub>, 47<sub>1</sub>, 47<sub>2</sub>, 47<sub>3</sub>には、それぞれ、トレリス上の遷移に基づいて、 $I\gamma$ 算出・記憶回路32により算出されたトレリス上の出力“000”、“001”、“010”、“011”、“100”、“101”、“110”、“111”に対応する枝の対数尤度 $I\gamma_t[000]$ ,  $I\gamma_t[001]$ ,  $I\gamma_t[010]$ ,  $I\gamma_t[011]$ ,  $I\gamma_t[100]$ ,  $I\gamma_t[101]$ ,  $I\gamma_t[110]$ ,  $I\gamma_t[111]$ と、各ステートにおける1時刻前の対数尤度 $I\alpha_{t-1}(0)$ ,  $I\alpha_{t-1}(1)$ ,  $I\alpha_{t-1}(2)$ ,  $I\alpha_{t-1}(3)$ が供給される。そして、加算比較選択回路47<sub>0</sub>, 47<sub>1</sub>, 47<sub>2</sub>, 47<sub>3</sub>は、それぞれ、次時刻のステート0、ステート1、ステート2、ステート3における対数尤度 $I\alpha$ を求める。

## 【0110】

具体的には、加算比較選択回路47<sub>0</sub>は、対数尤度 $I\gamma_t[000]$ ,  $I\gamma_t[101]$ ,  $I\gamma_t[111]$ ,  $I\gamma_t[010]$ を入力するとともに、対数尤度 $I\alpha_{t-1}(0)$ ,  $I\alpha_{t-1}(1)$ ,  $I\alpha_{t-1}(2)$ ,  $I\alpha_{t-1}(3)$ を入力し、ステート0における対数尤度 $I\alpha_t(0)$ を求める。

## 【0111】

また、加算比較選択回路47<sub>1</sub>は、対数尤度 $I\gamma_t[011]$ ,  $I\gamma_t[110]$ ,  $I\gamma_t[100]$ ,  $I\gamma_t[001]$ を入力するとともに、対数尤度 $I\alpha_{t-1}$

(0),  $I\alpha_{t-1}(1)$ ,  $I\alpha_{t-1}(2)$ ,  $I\alpha_{t-1}(3)$  を入力し、ステート1における対数尤度  $I\alpha_t(1)$  を求める。

【0112】

さらに、加算比較選択回路47<sub>2</sub>は、対数尤度  $I\gamma_t[101]$ ,  $I\gamma_t[000]$ ,  $I\gamma_t[010]$ ,  $I\gamma_t[111]$  を入力するとともに、対数尤度  $I\alpha_{t-1}(0)$ ,  $I\alpha_{t-1}(1)$ ,  $I\alpha_{t-1}(2)$ ,  $I\alpha_{t-1}(3)$  を入力し、ステート2における対数尤度  $I\alpha_t(2)$  を求める。

【0113】

さらにまた、加算比較選択回路47<sub>3</sub>は、対数尤度  $I\gamma_t[110]$ ,  $I\gamma_t[011]$ ,  $I\gamma_t[001]$ ,  $I\gamma_t[100]$  を入力するとともに、対数尤度  $I\alpha_{t-1}(0)$ ,  $I\alpha_{t-1}(1)$ ,  $I\alpha_{t-1}(2)$ ,  $I\alpha_{t-1}(3)$  を入力し、ステート3における対数尤度  $I\alpha_t(3)$  を求める。

【0114】

このような  $I\alpha$  算出回路43は、 $I\gamma$  算出・記憶回路32から供給された対数尤度  $I\gamma(\alpha)$  と、レジスタ42に保持されていた1時刻前の初期値  $I\alpha_0$  又は対数尤度  $I\alpha$  とを用いて、上式(27)に示した演算、すなわち、上式(28)に示した演算を行い、次時刻の各ステートにおける対数尤度  $I\alpha$  を算出する。 $I\alpha$  算出回路43は、算出した対数尤度  $I\alpha$  をセレクタ41に供給する。なお、加算比較選択回路47<sub>0</sub>, 47<sub>1</sub>, 47<sub>2</sub>, 47<sub>3</sub>については、後に詳述する。

【0115】

RAM44, 45は、それぞれ、コントローラ31から供給されたコントロール信号  $SC\alpha$  による制御の下に、レジスタ42から供給された対数尤度  $I\alpha(0)$ ,  $I\alpha(1)$ ,  $I\alpha(2)$ ,  $I\alpha(3)$  を順次保持する。ここで、対数尤度  $I\alpha(0)$ ,  $I\alpha(1)$ ,  $I\alpha(2)$ ,  $I\alpha(3)$  のビット数を、それぞれ、例えば8ビットとすると、RAM44, 45は、それぞれ、32ビットを1ワードとして、対数尤度  $I\alpha(0)$ ,  $I\alpha(1)$ ,  $I\alpha(2)$ ,  $I\alpha(3)$  を保持する。これらのRAM44, 45に保持された対数尤度  $I\alpha(0)$ ,  $I\alpha(1)$ ,  $I\alpha(2)$ ,  $I\alpha(3)$  は、選択回路46により所定の順序で読み出される。

【0116】

選択回路46は、コントローラ31から供給されたコントロール信号 $SC_{\alpha}$ による制御の下に、RAM44, 45から読み出した対数尤度 $I_{\alpha}(0)$ ,  $I_{\alpha}(1)$ ,  $I_{\alpha}(2)$ ,  $I_{\alpha}(3)$ を選択的に取り出し、対数尤度 $I_{\alpha}(\lambda)$ として軟出力算出回路35に供給する。

## 【0117】

このような $I_{\alpha}$ 算出・記憶回路33は、 $I_{\gamma}$ 算出・記憶回路32からの対数尤度 $I_{\gamma}(\alpha)$ の出力が開始される1時刻前の時点で初期化を行い、セクタ41により選択された初期値 $I_{\alpha_0}$ をレジスタ42に保持させる。そして、 $I_{\alpha}$ 算出・記憶回路33は、以後のクロック周期において、 $I_{\alpha}$ 算出回路43によって、 $I_{\gamma}$ 算出・記憶回路32から供給された対数尤度 $I_{\gamma}(\alpha)$ と、レジスタ42から供給された1時刻前の対数尤度 $I_{\alpha}$ とを用いて、次時刻における対数尤度 $I_{\alpha}$ を順次算出し、その対数尤度 $I_{\alpha}$ をレジスタ42に新たに保持させる。また、 $I_{\alpha}$ 算出・記憶回路33は、レジスタ42に保持された各ステートにおける対数尤度 $I_{\alpha}(0)$ ,  $I_{\alpha}(1)$ ,  $I_{\alpha}(2)$ ,  $I_{\alpha}(3)$ を順次RAM44, 45に保持させるとともに、選択回路46により所定の順序で読み出し、対数尤度 $I_{\alpha}(\lambda)$ として軟出力算出回路35に供給する。

## 【0118】

つぎに、 $I_{\beta}$ 算出・記憶回路34について説明する。 $I_{\beta}$ 算出・記憶回路34は、図7に示すように、各ステートにおける対数尤度 $I_{\beta}$ を算出する $I_{\beta}$ 算出回路51<sub>1</sub>, 51<sub>2</sub>と、算出された対数尤度 $I_{\beta}$ と対数尤度の初期値 $I_{\beta a}$ ,  $I_{\beta b}$ とのいずれか一方を選択するセクタ52<sub>1</sub>, 52<sub>2</sub>と、初期値 $I_{\beta a}$ ,  $I_{\beta b}$ 又は対数尤度 $I_{\beta}$ を保持するレジスタ53<sub>1</sub>, 53<sub>2</sub>と、これらのレジスタ53<sub>1</sub>, 53<sub>2</sub>から供給された対数尤度 $I_{\beta}$ を選択的に取り出す選択回路54とを有する。

## 【0119】

$I_{\beta}$ 算出回路51<sub>1</sub>, 51<sub>2</sub>は、それぞれ、図8に示すように、各ステートに応じた数、ここでは4つの加算比較選択回路55<sub>0</sub>, 55<sub>1</sub>, 55<sub>2</sub>, 55<sub>3</sub>を有する。

## 【0120】

加算比較選択回路  $55_0$ ,  $55_1$ ,  $55_2$ ,  $55_3$  には、それぞれ、トレリス上の遷移に基づいて、 $I_r$  算出・記憶回路 32 により算出されたトレリス上の出力 “000”、“001”、“010”、“011”、“100”、“101”、“110”、“111” に対応する枝の対数尤度  $I_{r_t}[000]$ ,  $I_{r_t}[001]$ ,  $I_{r_t}[010]$ ,  $I_{r_t}[011]$ ,  $I_{r_t}[100]$ ,  $I_{r_t}[101]$ ,  $I_{r_t}[110]$ ,  $I_{r_t}[111]$  と、各ステートにおける対数尤度  $I_{\beta_t}(0)$ ,  $I_{\beta_t}(1)$ ,  $I_{\beta_t}(2)$ ,  $I_{\beta_t}(3)$  が供給される。そして、加算比較選択回路  $55_0$ ,  $55_1$ ,  $55_2$ ,  $55_3$  は、それぞれ、1 時刻前のステート 0、ステート 1、ステート 2、ステート 3 における対数尤度  $I_{\beta}$  を求める。

## 【0121】

具体的には、加算比較選択回路  $55_0$  は、対数尤度  $I_{r_t}[000]$ ,  $I_{r_t}[011]$ ,  $I_{r_t}[101]$ ,  $I_{r_t}[110]$  を入力するとともに、対数尤度  $I_{\beta_t}(0)$ ,  $I_{\beta_t}(1)$ ,  $I_{\beta_t}(2)$ ,  $I_{\beta_t}(3)$  を入力し、ステート 0 における対数尤度  $I_{\beta_{t-1}}(0)$  を求める。

## 【0122】

また、加算比較選択回路  $55_1$  は、対数尤度  $I_{r_t}[101]$ ,  $I_{r_t}[110]$ ,  $I_{r_t}[000]$ ,  $I_{r_t}[011]$  を入力するとともに、対数尤度  $I_{\beta_t}(0)$ ,  $I_{\beta_t}(1)$ ,  $I_{\beta_t}(2)$ ,  $I_{\beta_t}(3)$  を入力し、ステート 1 における対数尤度  $I_{\beta_{t-1}}(1)$  を求める。

## 【0123】

さらに、加算比較選択回路  $55_2$  は、対数尤度  $I_{r_t}[111]$ ,  $I_{r_t}[100]$ ,  $I_{r_t}[010]$ ,  $I_{r_t}[001]$  を入力するとともに、対数尤度  $I_{\beta_t}(0)$ ,  $I_{\beta_t}(1)$ ,  $I_{\beta_t}(2)$ ,  $I_{\beta_t}(3)$  を入力し、ステート 2 における対数尤度  $I_{\beta_{t-1}}(2)$  を求める。

## 【0124】

さらにまた、加算比較選択回路  $55_3$  は、対数尤度  $I_{r_t}[010]$ ,  $I_{r_t}[001]$ ,  $I_{r_t}[111]$ ,  $I_{r_t}[100]$  を入力するとともに、対数尤度  $I_{\beta_t}(0)$ ,  $I_{\beta_t}(1)$ ,  $I_{\beta_t}(2)$ ,  $I_{\beta_t}(3)$  を入力し、ステート 3 における対数尤度  $I_{\beta_{t-1}}(3)$  を求める。



## 【0125】

このような $I\beta$ 算出回路 $51_1$ ,  $51_2$ は、それぞれ、 $I\gamma$ 算出・記憶回路32から供給された対数尤度 $I\gamma(\beta 1)$ ,  $I\gamma(\beta 2)$ と、レジスタ $53_1$ ,  $53_2$ に保持されていた初期値 $I\beta a$ ,  $I\beta b$ 又は対数尤度 $I\beta$ とを用いて、上式(29)に示した演算、すなわち、上式(30)に示した演算を行い、1時刻前の各ステートにおける対数尤度 $I\beta$ を算出する。ここで、対数尤度 $I\beta(0)$ ,  $I\beta(1)$ ,  $I\beta(2)$ ,  $I\beta(3)$ のビット数は、それぞれ、例えば8ビットであり、総ビット数は、32ビットとなる。 $I\beta$ 算出回路 $51_1$ ,  $51_2$ は、それぞれ、算出した対数尤度 $I\beta$ をセクタ $52_1$ ,  $52_2$ に供給する。なお、加算比較選択回路 $55_0$ ,  $55_1$ ,  $55_2$ ,  $55_3$ については、後に詳述する。

## 【0126】

セクタ $52_1$ ,  $52_2$ は、それぞれ、コントローラ31から供給されたコントロール信号 $SC\beta$ による制御の下に、初期化時には対数尤度の初期値 $I\beta a$ ,  $I\beta b$ を選択し、初期化時以外の時には $I\beta$ 算出回路 $51_1$ ,  $51_2$ のそれぞれから供給される対数尤度 $I\beta$ を選択する。なお、初期化は、 $I\gamma$ 算出・記憶回路32からの対数尤度 $I\gamma(\beta 1)$ ,  $I\gamma(\beta 2)$ の出力が開始される1時刻前の時点で行われ、以後打ち切り長の2倍の長さの周期毎に行われる。ここで、初期値 $I\beta a$ ,  $I\beta b$ としては、通常、全てのステートに対して例えば0や $\log(1/M)$ 、ここでは $\log(1/4)$ といったように、同じ値が与えられるが、終結された符号を復号する際には、終結するステートにおける値として $\log 1 = 0$ が与えられ、その他のステートにおける値として $\log 0 = -\infty$ が与えられる。セクタ $52_1$ ,  $52_2$ は、それぞれ、初期値 $I\beta a$ ,  $I\beta b$ 又は対数尤度 $I\beta$ のうちの選択した一方をレジスタ $53_1$ ,  $53_2$ に供給する。

## 【0127】

レジスタ $53_1$ ,  $53_2$ は、それぞれ、セクタ $52_1$ ,  $52_2$ から供給される初期値 $I\beta a$ ,  $I\beta b$ 又は対数尤度 $I\beta$ を保持する。そして、レジスタ $53_1$ ,  $53_2$ は、それぞれ、次時刻において、保持している初期値 $I\beta a$ ,  $I\beta b$ 又は対数尤度 $I\beta$ を $I\beta$ 算出回路 $51_1$ ,  $51_2$ 及び選択回路54に供給する。

## 【0128】

選択回路54は、コントローラ31から供給されたコントロール信号SC $\beta$ による制御の下に、レジスタ53<sub>1</sub>、53<sub>2</sub>のそれぞれから供給された対数尤度I $\beta$ (0)、I $\beta$ (1)、I $\beta$ (2)、I $\beta$ (3)を選択的に取り出し、対数尤度I $\beta$ ( $\lambda$ )として軟出力算出回路35に供給する。

## 【0129】

このようなI $\beta$ 算出・記憶回路34は、I $\gamma$ 算出・記憶回路32からの対数尤度I $\gamma$ ( $\beta$ 1)の出力が開始される1時刻前の時点及び以後打ち切り長の2倍の長さの周期毎に初期化を行い、セクタ52<sub>1</sub>により選択された初期値I $\beta$ aをレジスタ53<sub>1</sub>に保持させる。そして、I $\beta$ 算出・記憶回路34は、以後のクロック周期において、I $\beta$ 算出回路51<sub>1</sub>によって、I $\gamma$ 算出・記憶回路32から供給された対数尤度I $\gamma$ ( $\beta$ 1)と、レジスタ52<sub>1</sub>から供給された対数尤度I $\beta$ とを用いて、1時刻前における対数尤度I $\beta$ を順次算出し、その対数尤度I $\beta$ をレジスタ53<sub>1</sub>に新たに保持させる。

## 【0130】

また、I $\beta$ 算出・記憶回路34は、I $\gamma$ 算出・記憶回路32からの対数尤度I $\gamma$ ( $\beta$ 2)の出力が開始される1時刻前の時点及び以後打ち切り長の2倍の長さの周期毎に初期化を行い、セクタ52<sub>2</sub>により選択された初期値I $\beta$ bをレジスタ53<sub>2</sub>に保持させる。そして、I $\beta$ 算出・記憶回路34は、以後のクロック周期において、I $\beta$ 算出回路51<sub>2</sub>によって、I $\gamma$ 算出・記憶回路32から供給された対数尤度I $\gamma$ ( $\beta$ 2)と、レジスタ52<sub>2</sub>から供給された対数尤度I $\beta$ とを用いて、1時刻前における対数尤度I $\beta$ を順次算出し、その対数尤度I $\beta$ をレジスタ53<sub>2</sub>に新たに保持させる。そして、I $\beta$ 算出・記憶回路34は、レジスタ53<sub>1</sub>、53<sub>2</sub>のそれぞれに保持された各ステートにおける対数尤度I $\beta$ (0)、I $\beta$ (1)、I $\beta$ (2)、I $\beta$ (3)を選択回路54により所定の順序で読み出し、対数尤度I $\beta$ ( $\lambda$ )として軟出力算出回路35に供給する。

## 【0131】

さて、I $\alpha$ 算出・記憶回路33が有する加算比較選択回路47<sub>0</sub>、47<sub>1</sub>、47<sub>2</sub>、47<sub>3</sub>と、I $\beta$ 算出・記憶回路34が有する加算比較選択回路55<sub>0</sub>、55<sub>1</sub>、55<sub>2</sub>、55<sub>3</sub>について説明するが、これらの加算比較選択回路47<sub>0</sub>、47<sub>1</sub>、4

$7_2, 47_3, 55_0, 55_1, 55_2, 55_3$ は、それぞれ、入力と出力が異なるものの同一の構成からなる。そこで、以下では、加算比較選択回路  $47_0, 47_1, 47_2, 47_3, 55_0, 55_1, 55_2, 55_3$ を、加算比較選択回路 60 又は加算比較選択回路 120 と総称して説明する。また、以下では、加算比較選択回路  $47_0, 47_1, 47_2, 47_3$ のそれぞれに入力される4つの対数尤度  $I_\gamma$  及び加算比較選択回路  $55_0, 55_1, 55_2, 55_3$ のそれぞれに入力される4つの対数尤度  $I_\gamma$  を、 $IA, IB, IC, ID$  と総称し、加算比較選択回路  $47_0, 47_1, 47_2, 47_3$ のそれぞれに入力される4つの対数尤度  $I_\alpha$  及び加算比較選択回路  $55_0, 55_1, 55_2, 55_3$ のそれぞれに入力される4つの対数尤度  $I_\beta$  を、 $IE, IF, IG, IH$  と総称し、加算比較選択回路  $47_0, 47_1, 47_2, 47_3$ のそれぞれから出力される対数尤度  $I_\alpha$  及び加算比較選択回路  $55_0, 55_1, 55_2, 55_3$ のそれぞれから出力される対数尤度  $I_\beta$  を、 $IJ$  と総称して説明する。なお、以下の説明では、ハードウェアとして実装することを考慮して、確率を0以上の値で表し、確率が低いものほど大きな値で表すものとする。

#### 【0132】

まず、Log-BCJR アルゴリズムに基づいて実直に実装した加算比較選択回路 60 について説明する。この加算比較選択回路 60 は、図9に示すように、2つのデータを加算する加算器 61, 62, 69, 70 と、これらの加算器 61, 62 及び加算器 69, 70 からの出力の大小を比較する比較回路 63, 71 と、加算器 61, 62 及び加算器 69, 70 のそれぞれからの出力のいずれか一方を選択するセレクタ 64, 72 と、Log-BCJR アルゴリズムにおける補正項の値を算出する補正項算出回路 65, 73, 79 と、2つのデータの差分をとる差分器 66, 74, 80 と、差分器 66, 74 からの出力の大小を比較する比較回路 77 と、差分器 66, 74 のそれぞれからの出力のいずれか一方を選択するセレクタ 78 とを有する。

#### 【0133】

加算器 61 は、対数尤度  $IA, IE$  を入力し、これらの対数尤度  $IA, IE$  を加算する。例えば、加算比較選択回路 60 が加算比較選択回路  $47_0$  の場合には、加算器 61 は、対数尤度  $I_{\gamma_t}[000]$ 、対数尤度  $I_{\alpha_{t-1}}(0)$  を入力し、

これらの対数尤度  $I \gamma_t [000]$  , 対数尤度  $I \alpha_{t-1} (0)$  を加算する。加算器 61 は、加算して得られたデータを比較回路 63、セクタ 64 及び補正項算出回路 65 に供給する。なお、以下では、加算器 61 から出力されるデータを P として説明する。

## 【0134】

加算器 62 は、対数尤度  $I B$  ,  $I F$  を入力し、これらの対数尤度  $I B$  ,  $I F$  を加算する。例えば、加算比較選択回路 60 が加算比較選択回路 47<sub>0</sub> の場合には、加算器 62 は、対数尤度  $I \gamma_t [101]$  , 対数尤度  $I \alpha_{t-1} (1)$  を入力し、これらの対数尤度  $I \gamma_t [101]$  , 対数尤度  $I \alpha_{t-1} (1)$  を加算する。加算器 62 は、加算して得られたデータを比較回路 63、セクタ 64 及び補正項算出回路 65 に供給する。なお、以下では、加算器 62 から出力されるデータを Q として説明する。

## 【0135】

比較回路 63 は、加算器 61 から供給されたデータ P の値と、加算器 62 から供給されたデータ Q の値との大小を比較する。比較回路 63 は、比較結果を示す比較結果情報をセクタ 64 に供給する。

## 【0136】

セクタ 64 は、比較回路 63 から供給された比較結果情報に基づいて、加算器 61 から供給されたデータ P と、加算器 62 から供給されたデータ Q とのうち、値が小さいもの、すなわち、確率が高いものを選択する。セクタ 64 は、選択したデータを差分器 66 に供給する。

## 【0137】

補正項算出回路 65 は、加算器 61 から供給されたデータ P と、加算器 62 から供給されたデータ Q との差分値の絶対値を算出する絶対値算出回路 67 と、この絶対値算出回路 67 により算出された絶対値を用いて補正項を線形近似により算出する線形近似手段である線形近似回路 68 とを有する。補正項算出回路 65 は、 $\text{Log-BCJR}$  アルゴリズムにおける補正項の値、すなわち、上式 (28) 又は上式 (30) における右辺第 2 項の値を算出する。具体的には、補正項算出回路 65 は、補正項を、変数  $|P - Q|$  に対する 1 次元の関数で表し、この関

数の傾きを表す係数 $-a_1$  ( $a_1 > 0$ ) と、関数の切片を表す係数 $b_1$ とを用いて $-a_1 |P-Q| + b_1$ の形に線形近似した値を算出する。補正項算出回路65は、算出して得られたデータ $Z_1$ を差分器66に供給する。

## 【0138】

差分器66は、セレクタ64により選択されたデータと、補正項算出回路65から供給されたデータ $Z_1$ との差分値を求め、この差分値を比較回路77、セレクタ78及び補正項算出回路79に供給する。なお、以下では、差分器66から出力されるデータをTとして説明する。

## 【0139】

加算器69は、対数尤度IC, IGを入力し、これらの対数尤度IC, IGを加算する。例えば、加算比較選択回路60が加算比較選択回路470の場合には、加算器69は、対数尤度 $I\gamma_t[111]$ , 対数尤度 $I\alpha_{t-1}(2)$ を入力し、これらの対数尤度 $I\gamma_t[111]$ , 対数尤度 $I\alpha_{t-1}(2)$ を加算する。加算器69は、加算して得られたデータを比較回路71、セレクタ72及び補正項算出回路73に供給する。なお、以下では、加算器69から出力されるデータをRとして説明する。

## 【0140】

加算器70は、対数尤度ID, IHを入力し、これらの対数尤度ID, IHを加算する。例えば、加算比較選択回路60が加算比較選択回路470の場合には、加算器70は、対数尤度 $I\gamma_t[010]$ , 対数尤度 $I\alpha_{t-1}(3)$ を入力し、これらの対数尤度 $I\gamma_t[010]$ , 対数尤度 $I\alpha_{t-1}(3)$ を加算する。加算器70は、加算して得られたデータを比較回路71、セレクタ72及び補正項算出回路73に供給する。なお、以下では、加算器70から出力されるデータをSとして説明する。

## 【0141】

比較回路71は、加算器69から供給されたデータRの値と、加算器70から供給されたデータSの値との大小を比較する。比較回路71は、比較結果を示す比較結果情報をセレクタ72に供給する。

## 【0142】

セクタ72は、比較回路71から供給された比較結果情報に基づいて、加算器69から供給されたデータRと、加算器70から供給されたデータSとのうち、値が小さいもの、すなわち、確率が高いものを選択する。セクタ72は、選択したデータを差分器74に供給する。

## 【0143】

補正項算出回路73は、加算器69から供給されたデータRと、加算器70から供給されたデータSとの差分値の絶対値を算出する絶対値算出回路75と、この絶対値算出回路75により算出された絶対値を用いて補正項を線形近似により算出する線形近似手段である線形近似回路76とを有する。補正項算出回路73は、Log-BCJRアルゴリズムにおける補正項、すなわち、上式(28)又は上式(30)における右辺第2項の値を算出する。具体的には、補正項算出回路73は、補正項を、変数 $|R-S|$ に対する1次元の関数で表し、この関数の傾きを表す係数 $-a_2$  ( $a_2 > 0$ ) と、関数の切片を表す係数 $b_2$  とを用いて $-a_2 |R-S| + b_2$  の形に線形近似した値を算出する。補正項算出回路73は、算出して得られたデータ $Z_2$ を差分器74に供給する。

## 【0144】

差分器74は、セクタ72により選択されたデータと、補正項算出回路73から供給されたデータ $Z_2$ との差分値を求め、この差分値を比較回路77、セクタ78及び補正項算出回路79に供給する。なお、以下では、差分器74から出力されるデータをUとして説明する。

## 【0145】

比較回路77は、差分器66から供給されたデータTの値と、差分器74から供給されたデータUの値との大小を比較する。比較回路77は、比較結果を示す比較結果情報をセクタ78に供給する。

## 【0146】

セクタ78は、比較回路77から供給された比較結果情報に基づいて、差分器66から供給されたデータTと、差分器74から供給されたデータUとのうち、値が小さいもの、すなわち、確率が高いものを選択する。セクタ78は、選択したデータを差分器80に供給する。なお、このセクタ78により選択され

たデータは、上式(28)又は上式(30)における右辺第1項を示すものに他ならない。

## 【0147】

補正項算出回路79は、差分器66から供給されたデータTと、差分器74から供給されたデータUとの差分値の絶対値を算出する絶対値算出回路81と、この絶対値算出回路81により算出された絶対値を用いて補正項を線形近似により算出する線形近似回路82とを有する。補正項算出回路79は、Log-BCJRアルゴリズムにおける補正項、すなわち、上式(28)又は上式(30)における右辺第2項の値を算出する。具体的には、補正項算出回路79は、補正項を、変数 $|T-U|$ に対する1次元の関数で表し、この関数の傾きを表す係数 $-a_3$  ( $a_3 > 0$ )と、関数の切片を表す係数 $b_3$ とを用いて $-a_3|T-U| + b_3$ の形に線形近似した値を算出する。補正項算出回路79は、算出して得られたデータ $Z_3$ を差分器80に供給する。

## 【0148】

差分器80は、セレクタ78により選択されたデータと、補正項算出回路79から供給されたデータ $Z_3$ との差分値を求め、この差分値を対数尤度 $I_J$ として出力する。例えば、加算比較選択回路60が加算比較選択回路47<sub>0</sub>の場合には、差分器80は、対数尤度 $I_{\alpha_t}(0)$ を出力する。

## 【0149】

ここで、このような加算比較選択回路60における遅延量を見積もることを考える。なお、比較回路による遅延量及び差分器による遅延量は、ともに、加算器61、62のような通常の加算器による遅延量と同じものとする。

## 【0150】

加算比較選択回路60において回避できない遅延量は、同図から明らかなように、1つの加算器に相当するものとして、加算器61、62、69、70による遅延量と、比較回路63、71による遅延量と、差分器66、74による遅延量と、比較回路77による遅延量と、差分器80による遅延量とがある。また、加算比較選択回路60において回避できない遅延量は、1つのセレクタに相当するものとして、セレクタ64、72による遅延量と、セレクタ78による遅延量と

がある。すなわち、加算比較選択回路60は、少なくとも5つの加算器分の遅延量と、2つのセクタ分の遅延量とを有することになる。さらに、加算比較選択回路60には、補正項算出回路65, 73, 79による遅延量加わることになる。そこで、補正項算出回路65, 73, 79による遅延量を見積もることを考える。

## 【0151】

まず、絶対値算出回路67, 75, 81による遅延量を見積もる。なお、絶対値算出回路67, 75, 81は、それぞれ、同様の構成を有することから、ここでは、絶対値算出回路67, 75, 81を絶対値算出回路90と総称して説明する。また、ここでは、絶対値算出回路67に入力される2つのデータP, Q、絶対値算出回路75に入力される2つのデータR, S、及び絶対値算出回路81に入力される2つのデータT, Uを、PP, QQと総称し、絶対値算出回路67から出力される絶対値データ $|P - Q|$ 、絶対値算出回路75から出力される絶対値データ $|R - S|$ 、及び絶対値算出回路81から出力される絶対値データ $|T - U|$ を、 $|PP - QQ|$ と総称して説明する。

## 【0152】

絶対値算出回路90は、例えば図10に示すように、前段の2つの加算器からの出力の大小を比較する比較回路91と、2つのデータの差分をとる2つの差分器92, 93と、これらの差分器92, 93のそれぞれからの出力のいずれか一方を選択するセクタ94とを有するものとして実装することができる。

## 【0153】

すなわち、絶対値算出回路90は、比較回路91によって、前段の一方の加算器から供給されたデータPPの値と、前段の他方の加算器から供給されたデータQQの値との大小を比較する。これと同時に、絶対値算出回路90は、差分器92によって、データPPとデータQQとの差分値 $(PP - QQ)$ を求めるとともに、差分器93によって、データQQとデータPPとの差分値 $(QQ - PP)$ を求める。そして、絶対値算出回路90は、セクタ94によって、比較回路91による比較結果を示す比較結果情報に基づいて、差分値 $(PP - QQ)$ と差分値 $(QQ - PP)$ とのうち、正の値を有するものを選択し、選択した差分値を絶対



値データ  $|PP - QQ|$  として後段の線形近似回路に供給する。

【0154】

このような絶対値算出回路90は、比較回路91による処理と、差分器92、93による処理とが並列的に行われることから、1つの加算器と1つのセレクタ分の遅延量を有するものと見積もることができる。

【0155】

つぎに、線形近似回路68、76、82による遅延量を見積もる。なお、線形近似回路68、76、82は、それぞれ、同様の構成を有することから、ここでは、線形近似回路68、76、82を線形近似回路100と総称して説明する。また、ここでは、線形近似回路68に入力される絶対値データ  $|P - Q|$ 、線形近似回路76に入力される絶対値データ  $|R - S|$ 、及び線形近似回路82に入力される絶対値データ  $|T - U|$  を、 $|PP - QQ|$  と総称し、線形近似回路68から出力されるデータ  $Z_1$ 、線形近似回路76から出力されるデータ  $Z_2$ 、及び線形近似回路82から出力されるデータ  $Z_3$  を、 $Z$  と総称して説明する。

【0156】

線形近似回路100は、図11に示すように、曲線Cに示す関数  $F = \log \{ 1 + e^{(-|PP - QQ|)} \}$  を直線  $L_1$ 、 $L_2$  に示すような関数  $F = -a|PP - QQ| + b$  で近似するいわゆる線形近似法による  $\log - \text{sum}$  補正を行い、補正項の値を算出する。このとき、線形近似回路100は、少なくとも変数  $|PP - QQ|$  に乗算すべき関数  $F = -a|PP - QQ| + b$  の傾きを表す係数  $-a$  を2のべき乗を用いて表現して補正項の値を算出する。ここで、同図を参照すると、2のべき乗を用いて表現される係数  $-a$  としては、直線  $L_1$  に示すように、 $-a = -2^{-1} = -0.5$  とするか、若しくは、直線  $L_2$  に示すように、 $-a = -2^{-2} = -0.25$  とすることが考えられる。ここでは、 $-a = -0.25$  として説明する。この場合、線形近似回路100は、図12に示すように、差分器101と、比較回路102と、セレクタ103とを有するものとして実装することができる。

【0157】

差分器101は、関数  $F = -a|PP - QQ| + b$  の切片を表す係数  $b$  と、絶

対値算出回路90から供給された $n$ ビットからなる絶対値データ $|PP-QQ|$ のうちの上位 $n-2$ ビットとの差分値を求め、この差分値をセレクタ103に供給する。

## 【0158】

比較回路102は、係数 $b$ の値と、絶対値データ $|PP-QQ|$ のうちの上位 $n-2$ ビットで表されるデータ $|PP-QQ|[n:3]$ の値との大小を比較する。比較回路102は、比較結果を示す比較結果情報をセレクタ103に供給する。

## 【0159】

セレクタ103は、比較回路102から供給された比較結果情報に基づいて、差分器101から供給されたデータと、“0”とのうち、いずれか一方を選択する。具体的には、セレクタ103は、比較回路102による比較の結果、 $|PP-QQ|[n:3] \leq b$ であった場合には、差分器101から供給されたデータを選択し、 $|PP-QQ|[n:3] > b$ であった場合には、“0”を選択する。セレクタ103は、選択したデータを補正項を示すデータ $Z$ として、後段の差分器に供給する。

## 【0160】

このような線形近似回路100は、絶対値算出回路90から供給された $n$ ビットからなる絶対値データ $|PP-QQ|$ の下位1ビット目から下位2ビット目までを切り捨て、残りの上位 $n-2$ ビットで表されるデータを係数 $b$ から差分する。すなわち、線形近似回路100は、絶対値データ $|PP-QQ|$ のうちの下位2ビットを切り捨ててビットシフトすることによって、 $|PP-QQ|$ を $1/4 = 0.25$ 倍することができ、残りの上位 $n-2$ ビットで表されるデータを係数 $b$ から差分することによって、結果として $-0.25|PP-QQ| + b$ の演算を実現することができる。

## 【0161】

また、線形近似回路100は、補正項が正值であることから、比較回路102による比較の結果、差分器101から出力されたデータの値が負であった場合、すなわち、補正項が負値として算出された場合には、セレクタ103により“0

”を出力することによって、補正項が負値をとることを回避することができる。

【0162】

なお、例えば $-a = -2^{-1} = -0.5$ とした場合には、線形近似回路100は、絶対値データ $|PP-QQ|$ の下位1ビットを切り捨ててビットシフトすればよく、係数 $-a$ を表現するべき数に応じて、絶対値データ $|PP-QQ|$ の下位ビットから切り捨てるようにすればよい。

【0163】

このような線形近似回路100は、実際の乗算器を不要とし、差分器101による処理と、比較回路102による処理とが並列的に行われることから、1つの加算器と1つのセクタ分の遅延量を有するものと見積もることができる。

【0164】

さらに、線形近似回路100としては、係数 $b$ をも2のべき乗を用いて表現して補正項の値を算出するものも考えられる。すなわち、係数 $b$ を $2^m - 1$ で表現される値とし、 $-a = -0.25$ とした場合、線形近似回路100は、図13に示すように、インバータ111と、ORゲート112と、セクタ113とを有するものとして実装することができる。

【0165】

インバータ111は、絶対値算出回路90から供給された $n$ ビットからなる絶対値データ $|PP-QQ|$ のうちの低位3ビット目から低位 $m+2$ ビット目までの $m$ ビットを反転する。インバータ111は、反転して得られたデータをセクタ113に供給する。

【0166】

ORゲート112は、絶対値算出回路90から供給された $n$ ビットからなる絶対値データ $|PP-QQ|$ のうちの低位 $m+3$ ビット目から $n$ ビット目までの上位 $n-m-2$ ビットの論理和をとる。ORゲート112は、求めた論理和をセクタ113に供給する。

【0167】

セクタ113は、ORゲート112から供給された論理和に基づいて、インバータ111から供給されたデータと、“0”とのうち、いずれか一方を選択す

る。具体的には、セクタ113は、ORゲート112から供給された論理和が“0”であった場合には、インバータ111から供給されたデータを選択し、ORゲート112から供給された論理和が“1”であった場合には、“0”を選択する。セクタ113は、選択したデータを補正項を示すデータZとして、後段の差分器に供給する。

## 【0168】

このような線形近似回路100は、絶対値算出回路90から供給されたnビットからなる絶対値データ $|PP-QQ|$ の下位1ビット目から下位2ビット目までを切り捨て、残りの上位 $n-2$ ビットのうちの下位3ビット目から下位 $m+2$ ビット目までのmビットをインバータ111により反転する。これと同時に、線形近似回路100は、ORゲート112によって、下位 $m+3$ ビット目からnビット目までの $n-m-2$ ビットの論理和をとる。

## 【0169】

すなわち、線形近似回路100は、上述したように、絶対値データ $|PP-QQ|$ のうちの下位2ビットを切り捨ててビットシフトすることによって、 $|PP-QQ|$ を $1/4=0.25$ 倍することができる。したがって、線形近似回路100は、絶対値データ $|PP-QQ|$ のうちの上位 $n-2$ ビットで表されるデータ $|PP-QQ|[n:3]$ 、すなわち、 $0.25|PP-QQ|$ を $-1$ 倍し、この値に $2^m-1$ で表現される係数bを加算すればよい。

## 【0170】

ここで、線形近似回路100による演算を論理式で表現するために、図14(A)に示すように、nビットからなる絶対値データ $|PP-QQ|$ の下位2ビットを切り捨てて得られる $0.25|PP-QQ|$ を $A=(A_n, A_{n-1}, \dots, A_{m+3}, A_{m+2}, \dots, A_3)$ とし、残りの上位 $n-2$ ビットのうちの下位3ビット目から下位 $m+2$ ビット目までのmビットと、下位 $m+3$ ビット目からnビット目までの $n-m-2$ ビットとを、それぞれ、 $A'$ 、 $A''$ として説明する。

## 【0171】

まず、線形近似回路100により求めるべき $-0.25|PP-QQ|+2^m-1=-A+(2^m-1)$ が負値をとる場合を考える。この場合、次式(32

）に示す同値関係が成立する。すなわち、“ $-A + (2^m - 1)$ ”が負値をとる場合には、 $A'$ が正值をとることになる。換言すれば、“ $-A + (2^m - 1)$ ”が負値をとる場合には、 $A'$ を構成する全てのビットの論理和が“1”となることになる。

【0172】

【数32】

$$\begin{aligned} -A + (2^m - 1) < 0 &\Leftrightarrow A > 2^m - 1 \\ &\Leftrightarrow A' > 0 \end{aligned} \quad (32)$$

【0173】

一方、線形近似回路100により求めるべき“ $-0.25 | PP - QQ | + 2^m - 1 = -A + (2^m - 1)$ ”が0以上の値をとる場合を考える。この場合、上式(32)に示した同値関係より $A' = 0$ であることから、次式(33)が成立する。

【0174】

【数33】

$$-A + (2^m - 1) = -A' + (2^m - 1) \quad (33)$$

【0175】

ここで、 $2^m - 1$ は、 $m$ ビット全てが“1”であるデータであることに着目すると、“ $-A' + (2^m - 1)$ ”は、同図(B)に示すように、 $A'$ の否定で表される。

## 【0176】

以上の議論から、線形近似回路100は、Aの下位mビットの否定を求めればよいことになる。したがって、線形近似回路100は、絶対値データ $|PP-QQ|$ のうちの下位3ビット目から下位 $m+2$ ビット目までのmビットで表されるデータ $|PP-QQ|[m+2:2]$ をインバータ111により反転することによって、“ $-0.25|PP-QQ|+2^m-1$ ”の演算を実現することができる。

## 【0177】

また、線形近似回路100は、ORゲート112によって、絶対値データ $|PP-QQ|$ のうちの下位 $m+3$ ビット目からnビット目までの $n-m-2$ ビットで表されるデータ $|PP-QQ|[m+3:n]$ の論理和をとることによって、“ $-0.25|PP-QQ|+2^m-1$ ”の値の正負を判断することができる。そのため、線形近似回路100は、補正項が正值であることから、ORゲート112による論理和が“1”であった場合、すなわち、補正項が負値として算出された場合には、セレクタ113により“0”を出力することによって、補正項が負値をとることを回避することができる。

## 【0178】

なお、線形近似回路100は、係数aを $-2^{-k}$ で表現したとき、絶対値データ $|PP-QQ|$ の下位1ビット目から下位kビット目までを切り捨ててビットシフトし、下位 $k+1$ ビット目から下位 $m+k$ ビット目までのmビットを反転することになる。例として、 $n=5$ 、 $m=2$ の場合、すなわち、“ $-0.25|PP-QQ|+3$ ”の演算を行う場合において、絶対値データ $|PP-QQ|$ とデータZとの関係を求めると、次表1に示すようになる。なお、同表には、上述したA'の否定、すなわち、インバータ111から出力されるデータについても示している。

## 【0179】

【表 1】

表 1 絶対値データ  $|P-Q|$  とデータ  $Z$  との関係

	$ P-Q $	$\overline{A}$	$Z$
31	11111	00	0
30	11110	00	0
29	11101	00	0
28	11100	00	0
27	11011	01	0
26	11010	01	0
25	11001	01	0
24	11000	01	0
23	10111	10	0
22	10110	10	0
21	10101	10	0
20	10100	10	0
19	10011	11	0
18	10010	11	0
17	10001	11	0
16	10000	11	0
15	01111	00	0
14	01110	00	0
13	01101	00	0
12	01100	00	0
11	01011	01	1
10	01010	01	1
9	01001	01	1
8	01000	01	1
7	00111	10	2
6	00110	10	2
5	00101	10	2
4	00100	10	2
3	00011	11	3
2	00010	11	3
1	00001	11	3
0	00000	11	3

## 【0180】

同表に示すように、線形近似回路100は、絶対値データ $|PP-QQ|$ が0乃至12の範囲では、絶対値データ $|PP-QQ|$ のうちの下位3ビット目から下位 $2+2=4$ ビット目までの2ビットで表されるデータ $|PP-QQ|[4:2]$ をインバータ111により反転したものをデータZとして出力し、絶対値データ $|PP-QQ|$ が13以上の範囲では、インバータ111からの出力が負となることから、0を出力する。

## 【0181】

このような線形近似回路100は、実際の乗算器及び加算器を不要とし、ビットシフトとインバータのみで構成することができ、1つのセクタ分の遅延量を有するものと見積もることができる。

## 【0182】

以上のように、補正項算出回路65, 73, 79による遅延量は、線形近似回路68, 76, 82を図12に示すように構成することによって、2つの加算器と2つのセクタ分として見積もることができ、線形近似回路68, 76, 82を図13に示すように構成することによって、1つの加算器と2つのセクタ分にまで抑えることができる。

## 【0183】

したがって、加算比較選択回路60における遅延量は、比較回路63, 71による処理と、絶対値算出回路67, 75による処理とが並列的に行われるとともに、比較回路77による処理と、絶対値算出回路81による処理とが並列的に行われることから、線形近似回路68, 76, 82を図12に示すように構成した場合には、7つの加算器と4つのセクタ分となり、線形近似回路68, 76, 82を図13に示すように構成した場合には、5つの加算器と4つのセクタ分となる。

## 【0184】

ここで、加算比較選択回路60による遅延量をさらに小さくすることを考える。加算比較選択回路60における処理の高速化を実現するために、尤度の高い少なくとも2つ以上のパスを用いて補正を行う。以下、この加算比較選択回路を上



述した加算比較選択回路120と総称して説明する。なお、ここでは、加算比較選択回路120は、最も尤度の高いパスと2番目に尤度の高いパスとの間だけで補正を行うものとする。

## 【0185】

まず、補正を考える前に、最も尤度の高いパスを高速に選択することを考える。なお、以下の説明では、最も尤度の高いパスを最尤パスと称し、2番目に尤度の高いパスを準最尤パスと称するものとする。

## 【0186】

加算比較選択回路120は、データP、データQ、データR及びデータSのうち、データPの値とデータQの値との大小を比較するとともに、データRの値とデータSの値との大小を比較する。また、加算比較選択回路120は、これと同時に、4つのデータの全てについて、2つのデータの組み合わせ値の大小を比較し、データPの値とデータQの値との大小の比較結果を示す比較結果情報と、データRの値とデータSの値との大小の比較結果を示す比較結果情報とに基づいて、選択するパスを決定する。

## 【0187】

具体的には、加算比較選択回路120は、このようにパスを選択するための回路として、図15に示すパス選択手段であるパス選択部130を有することにより実装することができる。パス選択部130は、比較手段である6つの比較回路131、132、133、134、135、136と、6つのセレクタ137、138、139、140、141、142とを有する。

## 【0188】

比較回路131は、上述した比較回路63に対応するものであり、データPの値とデータQの値との大小を比較する。比較回路131は、比較結果を示す比較結果情報 $C_1$ をセレクタ137、139、140に供給する。

## 【0189】

比較回路132は、上述した比較回路71に対応するものであり、データRの値とデータSの値との大小を比較する。比較回路132は、比較結果を示す比較結果情報 $C_2$ をセレクタ138、141に供給する。

## 【0190】

比較回路133は、データPの値とデータRの値との大小を比較する。比較回路133は、比較結果を示す比較結果情報をセクタ139に供給する。

## 【0191】

比較回路134は、データQの値とデータRの値との大小を比較する。比較回路134は、比較結果を示す比較結果情報をセクタ139に供給する。

## 【0192】

比較回路135は、データPの値とデータSの値との大小を比較する。比較回路135は、比較結果を示す比較結果情報をセクタ140に供給する。

## 【0193】

比較回路136は、データQの値とデータSの値との大小を比較する。比較回路136は、比較結果を示す比較結果情報をセクタ140に供給する。

## 【0194】

セクタ137は、上述したセクタ64に対応するものであり、比較回路131から供給された比較結果情報 $C_1$ に基づいて、データPとデータQとのうち、値が小さいもの、すなわち、確率が高いものを選択する。セクタ137は、選択したデータをセクタ142に供給する。

## 【0195】

セクタ138は、上述したセクタ72に対応するものであり、比較回路132から供給された比較結果情報 $C_2$ に基づいて、データRとデータSとのうち、値が小さいもの、すなわち、確率が高いものを選択する。セクタ138は、選択したデータをセクタ142に供給する。

## 【0196】

セクタ139は、比較回路131から供給された比較結果情報 $C_1$ に基づいて、比較回路133から供給される比較結果情報と、比較回路134から供給される比較結果情報とのうち、いずれか一方の比較結果情報を選択する。具体的には、セクタ139は、比較回路131による比較の結果、 $P \leq Q$ の場合には、比較回路133から供給される比較結果情報を選択し、 $P > Q$ の場合には、比較回路134から供給される比較結果情報を選択する。セクタ139は、選択し

た比較結果情報をセクタ141に供給する。

【0197】

セクタ140は、比較回路131から供給された比較結果情報 $C_1$ に基づいて、比較回路135から供給される比較結果情報と、比較回路136から供給される比較結果情報とのうち、いずれか一方の比較結果情報を選択する。具体的には、セクタ140は、比較回路131による比較の結果、 $P \leq Q$ の場合には、比較回路135から供給される比較結果情報を選択し、 $P > Q$ の場合には、比較回路136から供給される比較結果情報を選択する。セクタ140は、選択した比較結果情報をセクタ141に供給する。

【0198】

セクタ141は、比較回路132から供給された比較結果情報 $C_2$ に基づいて、セクタ139から供給される比較結果情報と、セクタ140から供給される比較結果情報とのうち、いずれか一方の比較結果情報を選択する。具体的には、セクタ141は、比較回路132による比較の結果、 $R \leq S$ の場合には、セクタ139から供給される比較結果情報を選択し、 $R > S$ の場合には、セクタ140から供給される比較結果情報を選択する。セクタ141は、選択した比較結果情報 $C_3$ をセクタ142に供給する。

【0199】

セクタ142は、上述したセクタ78に対応するものであり、セクタ141から供給された比較結果情報 $C_3$ に基づいて、セクタ137から供給されるデータと、セクタ138から供給されるデータとのうち、いずれか一方のデータを選択して出力する。具体的には、セクタ142は、セクタ141から供給される比較結果選択情報 $C_3$ がセクタ139から供給される比較選択情報である場合には、セクタ137から供給されるデータを選択して出力し、セクタ141から供給される比較結果選択情報 $C_3$ がセクタ140から供給される比較選択情報である場合には、セクタ138から供給されるデータを選択して出力する。

【0200】

このようなパス選択部130は、各ステートに到達した4つのパスに対応する

データであるデータP、データQ、データR及びデータSの中から選択した2つのパスに対応するデータの組み合わせの全てについて尤度の大小を比較することによって、これらのデータP、データQ、データR及びデータSのうち、尤度の高い少なくとも2つ以上のパスに対応するデータを求め、これらのパスに対応するデータの中から、最も尤度の高いパスである最尤パスに対応するデータを選択する。より具体的には、パス選択部130は、データP、データQ、データR及びデータSについて、いわば勝ち抜き戦に喩えられる動作を行うことによって、データPの値、データQの値、データRの値及びデータSの値の大小を比較し、最尤パスに対応するデータを選択する。

## 【0201】

すなわち、パス選択部130は、図16(A)に示すように、比較回路131によって、データPとデータQとの間で1回戦を行うとともに、比較回路132によって、データRとデータSとの間で1回戦を行う。これと同時に、パス選択部130は、比較回路133、比較回路134、比較回路135及び比較回路136によって、同図(B)乃至同図(E)に示すように、データP、データQ、データR及びデータSの間で行われる可能性のある4通りの決勝戦を行う。そして、パス選択部130は、比較回路131及び比較回路132による1回戦の結果を示す比較結果情報 $C_1$ 、 $C_2$ に基づいて、4通りの決勝戦の可能性の中から最も高い確率で生じる組み合わせを示す比較選択情報 $C_3$ を決定し、この比較選択情報 $C_3$ に基づいて、データP、データQ、データR及びデータSの中から最尤パスに対応するデータを選択する。

## 【0202】

つぎに、最尤パスと準最尤パスとの間だけで補正を行うために選択すべき最尤パスに対応するデータと準最尤パスに対応するデータとの差分値の絶対値を求めることを考える。

## 【0203】

補正のために選択すべき準最尤パスに対応するデータは、最尤パスに対応するデータの1回戦における相手と決勝戦における相手とのうち、値が小さいもの、すなわち、確率が高いものとなる。例えば、図17に示すようにパス選択部13

0による勝ち抜き戦の結果が得られた場合、すなわち、比較結果情報 $C_1$ ,  $C_2$ ,  $C_3$ が、それぞれ、 $P \leq Q$ ,  $R \leq S$ ,  $P \leq R$ を示すものであった場合には、加算比較選択回路120は、絶対値データ $|P - Q|$ ,  $|P - R|$ のうちの値が小さい方を、補正を行うために選択すべき絶対値データとして選択すればよい。

## 【0204】

このように、2つの絶対値データを実直に比較して選択するものとする、加算比較選択回路120は、図18に示す絶対値選択手段である絶対値データ選択部150を有する実装となる。この絶対値データ選択部150は、2つの絶対値データを算出する2つの絶対値算出回路151, 152と、2つの絶対値データの大小を比較する比較回路153と、この比較回路153による比較結果を示す比較結果情報に基づいて、2つの絶対値データのいずれか一方を選択するセクタ154とを有するものとなる。

## 【0205】

しかしながら、この絶対値データ選択部150は、絶対値算出回路151, 152による遅延量の他に、比較回路153による遅延量と、セクタ154による遅延量とを有することから、加算比較選択回路120には、絶対値データ選択部150を有する場合には、1つの加算器分の遅延量と、1つのセクタ分の遅延量とがさらに加わることになる。

## 【0206】

そこで、例えば、絶対値データ $|P - Q|$ ,  $|P - R|$ の大小関係は、データQ, Rの値の大小から求めることができることと、データQ, Rの値の大小関係は、パス選択部130により既に求められていることといったように、既に求められている2つのデータの大小関係に基づいて、絶対値データの大小関係を判別できることに着目する。

## 【0207】

この点に着目した場合、加算比較選択回路120は、図19に示す絶対値データ選択部160を有する実装とすることができる。

## 【0208】

絶対値データ選択部160は、同図に示すように、2つのデータの差分値の絶

対値を算出する6つの絶対値算出回路161, 162, 163, 164, 165, 166と、入力されるデータのうちのいずれか1つを選択する4つのセクタ167, 168, 169, 170とを有する。

【0209】

絶対値算出回路161は、データPとデータQとの差分値( $P-Q$ )を求めるとともに、データQとデータPとの差分値( $Q-P$ )を求め、これらの差分値( $P-Q$ )と差分値( $Q-P$ )とのうち、正の値を有するものを選択し、選択した差分値を絶対値データ $|P-Q|$ としてセクタ168に供給する。

【0210】

絶対値算出回路162は、データRとデータSとの差分値( $R-S$ )を求めるとともに、データSとデータRとの差分値( $S-R$ )を求め、これらの差分値( $R-S$ )と差分値( $S-R$ )とのうち、正の値を有するものを選択し、選択した差分値を絶対値データ $|R-S|$ としてセクタ168に供給する。

【0211】

絶対値算出回路163は、データPとデータRとの差分値( $P-R$ )を求めるとともに、データRとデータPとの差分値( $R-P$ )を求め、これらの差分値( $P-R$ )と差分値( $R-P$ )とのうち、正の値を有するものを選択し、選択した差分値を絶対値データ $|P-R|$ としてセクタ169に供給する。

【0212】

絶対値算出回路164は、データQとデータRとの差分値( $Q-R$ )を求めるとともに、データRとデータQとの差分値( $R-Q$ )を求め、これらの差分値( $Q-R$ )と差分値( $R-Q$ )とのうち、正の値を有するものを選択し、選択した差分値を絶対値データ $|Q-R|$ としてセクタ169に供給する。

【0213】

絶対値算出回路165は、データPとデータSとの差分値( $P-S$ )を求めるとともに、データSとデータPとの差分値( $S-P$ )を求め、これらの差分値( $P-S$ )と差分値( $S-P$ )とのうち、正の値を有するものを選択し、選択した差分値を絶対値データ $|P-S|$ としてセクタ169に供給する。

【0214】

絶対値算出回路166は、データQとデータSとの差分値( $Q-S$ )を求めるとともに、データSとデータQとの差分値( $S-Q$ )を求め、これらの差分値( $Q-S$ )と差分値( $S-Q$ )とのうち、正の値を有するものを選択し、選択した差分値を絶対値データ $|Q-S|$ としてセレクタ169に供給する。

## 【0215】

セレクタ167は、パス選択部130における比較回路133, 134, 135, 136から供給された4つの比較結果情報のうちのいずれか1つを選択する。セレクタ167は、選択した比較結果情報をセレクタ170に供給する。なお、ここでは、このセレクタ167により選択された比較結果情報を $C_4$ として識別するものとする。

## 【0216】

セレクタ168は、パス選択部130におけるセレクタ141から供給された比較結果情報 $C_3$ に基づいて、絶対値算出回路161, 162から供給された絶対値データ $|P-Q|$ ,  $|R-S|$ のうちのいずれか一方を選択する。セレクタ168は、選択した絶対値データをセレクタ170に供給する。

## 【0217】

セレクタ169は、パス選択部130における比較回路131, 132のそれぞれから供給された比較結果情報 $C_1$ ,  $C_2$ に基づいて、絶対値算出回路163, 164, 165, 166のそれぞれから供給された絶対値データ $|P-R|$ ,  $|Q-R|$ ,  $|P-S|$ ,  $|Q-S|$ のうちのいずれか1つを選択する。セレクタ168は、選択した絶対値データをセレクタ170に供給する。

## 【0218】

セレクタ170は、セレクタ167から供給された比較結果情報 $C_4$ に基づいて、セレクタ168から供給された絶対値データと、セレクタ169から供給された絶対値データとのうち、いずれか一方を選択し、補正項の算出に用いる絶対値データとして出力する。

## 【0219】

このような絶対値データ選択部160は、パス選択部130における比較回路133, 134, 135, 136により求められた比較結果情報に基づいて、絶

対値算出回路161, 162, 163, 164, 165, 166により算出された絶対値データの大小関係を判別することができ、最尤パスに対応するデータと準最尤パスに対応するデータとの差分値の絶対値を求めて選択することができる。絶対値データ選択部160においては、例えば、図17に示したようにパス選択部130による勝ち抜き戦の結果が得られた場合、すなわち、比較結果情報 $C_1$ ,  $C_2$ ,  $C_3$ が、それぞれ、 $P \leq Q$ ,  $R \leq S$ ,  $P \leq R$ を示すものであった場合には、セレクタ167により選択される比較結果情報 $C_4$ は、比較回路134から出力された比較結果情報となり、セレクタ168により選択される絶対値データは、絶対値算出回路161から出力された絶対値データ $|P - Q|$ となり、セレクタ169により選択される絶対値データは、絶対値算出回路163から出力された絶対値データ $|P - R|$ となる。したがって、絶対値データ選択部160は、セレクタ167により選択される比較結果情報 $C_4$ が $Q \leq R$ を示すものであった場合には、セレクタ170により絶対値データ $|P - Q|$ を選択して出力し、セレクタ167により選択される比較結果情報 $C_4$ が $Q > R$ を示すものであった場合には、セレクタ170により絶対値データ $|P - R|$ を選択して出力する。

#### 【0220】

このように絶対値データ選択部160は、上述したように、勝ち抜き戦の結果に応じて、セレクタ167により比較結果情報 $C_4$ を選択すればよい。ここでは、4つのデータP、データQ、データR及びデータSについての勝ち抜き戦であることから、絶対値データ選択部160は、図20に示すように、8つの勝ち抜き戦の結果に応じて、セレクタ167により比較結果情報 $C_4$ を選択することになる。

#### 【0221】

まず、同図(A)に示すように、データPが最尤パスに対応するデータとして選択され、○印が付与されたデータQ, Rのいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ167は、比較結果情報 $C_4$ として、比較回路134から出力された比較結果情報、すなわち、データQとデータRとの大小関係を示す比較結果情報を選択する。このとき、パス選択部130における比較結果情報 $C_1$ ,  $C_2$ ,  $C_3$ は、それぞれ、 $P \leq Q$ ,  $R \leq S$ ,  $P \leq R$ を示すも



のとなる。

【0222】

また、同図（B）に示すように、データRが最尤パスに対応するデータとして選択され、○印が付与されたデータP、Sのいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ167は、比較結果情報 $C_4$ として、比較回路135から出力された比較結果情報、すなわち、データPとデータSとの大小関係を示す比較結果情報を選択する。このとき、パス選択部130における比較結果情報 $C_1$ 、 $C_2$ 、 $C_3$ は、それぞれ、 $P \leq Q$ 、 $R \leq S$ 、 $P > R$ を示すものとなる。

【0223】

さらに、同図（C）に示すように、データPが最尤パスに対応するデータとして選択され、○印が付与されたデータQ、Sのいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ167は、比較結果情報 $C_4$ として、比較回路136から出力された比較結果情報、すなわち、データQとデータSとの大小関係を示す比較結果情報を選択する。このとき、パス選択部130における比較結果情報 $C_1$ 、 $C_2$ 、 $C_3$ は、それぞれ、 $P \leq Q$ 、 $R > S$ 、 $P \leq S$ を示すものとなる。

【0224】

さらにまた、同図（D）に示すように、データSが最尤パスに対応するデータとして選択され、○印が付与されたデータP、Rのいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ167は、比較結果情報 $C_4$ として、比較回路133から出力された比較結果情報、すなわち、データPとデータRとの大小関係を示す比較結果情報を選択する。このとき、パス選択部130における比較結果情報 $C_1$ 、 $C_2$ 、 $C_3$ は、それぞれ、 $P \leq Q$ 、 $R > S$ 、 $P > S$ を示すものとなる。

【0225】

また、同図（E）に示すように、データQが最尤パスに対応するデータとして選択され、○印が付与されたデータP、Rのいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ167は、比較結果情報 $C_4$ として、

比較回路 133 から出力された比較結果情報、すなわち、データ P とデータ R との大小関係を示す比較結果情報を選択する。このとき、パス選択部 130 における比較結果情報  $C_1$ ,  $C_2$ ,  $C_3$  は、それぞれ、 $P > Q$ ,  $R \leq S$ ,  $Q \leq R$  を示すものとなる。

## 【0226】

さらに、同図 (F) に示すように、データ R が最尤パスに対応するデータとして選択され、○印が付与されたデータ Q, S のいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ 167 は、比較結果情報  $C_4$  として、比較回路 136 から出力された比較結果情報、すなわち、データ Q とデータ S との大小関係を示す比較結果情報を選択する。このとき、パス選択部 130 における比較結果情報  $C_1$ ,  $C_2$ ,  $C_3$  は、それぞれ、 $P > Q$ ,  $R \leq S$ ,  $Q > R$  を示すものとなる。

## 【0227】

さらにまた、同図 (G) に示すように、データ Q が最尤パスに対応するデータとして選択され、○印が付与されたデータ P, S のいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ 167 は、比較結果情報  $C_4$  として、比較回路 135 から出力された比較結果情報、すなわち、データ P とデータ S との大小関係を示す比較結果情報を選択する。このとき、パス選択部 130 における比較結果情報  $C_1$ ,  $C_2$ ,  $C_3$  は、それぞれ、 $P > Q$ ,  $R > S$ ,  $Q \leq S$  を示すものとなる。

## 【0228】

また、同図 (H) に示すように、データ S が最尤パスに対応するデータとして選択され、○印が付与されたデータ Q, R のいずれか一方が準最尤パスに対応するデータとして選択された場合、セレクタ 167 は、比較結果情報  $C_4$  として、比較回路 134 から出力された比較結果情報、すなわち、データ Q とデータ R との大小関係を示す比較結果情報を選択する。このとき、パス選択部 130 における比較結果情報  $C_1$ ,  $C_2$ ,  $C_3$  は、それぞれ、 $P > Q$ ,  $R > S$ ,  $Q > S$  を示すものとなる。

## 【0229】

さて、このような選択動作を行うセクタ167として、通常のセクタを用いて実装した場合には、適切な比較結果情報を選択するための新たな勝ち抜き戦を行う必要が生じることから、絶対値データ選択部160は、遅延量の増加を招くことになる。

#### 【0230】

そこで、絶対値データ選択部160は、セクタ167の代わりに、以下に示す論理演算から導出される構成によって、セクタ167による選択動作を実現することができる。

#### 【0231】

まず、比較結果情報 $C_4$ に基づいてデータを選択するセクタ170は、比較結果情報 $C_4$ が $P \leq R$ ,  $Q \leq R$ ,  $P \leq S$ 又は $Q \leq S$ を示すものである場合には、セクタ168から供給されるデータ、すなわち、勝ち抜き戦における1回戦のデータを選択し、比較結果情報 $C_4$ が $P > R$ ,  $Q > R$ ,  $P > S$ 又は $Q > S$ を示すものである場合には、セクタ169から供給されるデータ、すなわち、勝ち抜き戦における決勝戦の可能性のあるデータを選択する。そこで、 $P \leq R$ ,  $Q \leq R$ ,  $P \leq S$ 又は $Q \leq S$ を示すものである場合における比較結果情報 $C_4$ を“1”と表し、セクタ168から供給されるデータ、すなわち、勝ち抜き戦における1回戦のデータを“ $\Delta_{1st}$ ”と表し、セクタ169から供給されるデータ、すなわち、勝ち抜き戦における決勝戦の可能性のあるデータを“ $\Delta_{final}$ ”と表すものとする、次式(34)に示す同値関係が成立する。なお、次式(34)における“ $\wedge$ ”は、論理積を示し、“ $\vee$ ”は、論理和を示し、“ $\neg$ ”は、否定を示す論理記号である。

#### 【0232】

【数 3 4】

$$\begin{aligned}
C_4 = 1 &\Leftrightarrow \Delta_{1st} \leq \Delta_{final} \\
&\Leftrightarrow \left\{ \min(P, Q) \leq \min(R, S) \wedge \max(P, Q) \leq \min(R, S) \right\} \\
&\vee \left\{ \min(R, S) \leq \min(P, Q) \wedge \max(R, S) \leq \min(P, Q) \right\} \\
&\Leftrightarrow \left\{ \max(P, Q) \leq \min(R, S) \right\} \vee \left\{ \max(R, S) \leq \min(P, Q) \right\} \\
&\Leftrightarrow (P \leq R \wedge Q \leq R \wedge P \leq S \wedge Q \leq S) \\
&\quad \vee (R \leq P \wedge R \leq Q \wedge S \leq P \wedge S \leq Q) \\
&\Leftrightarrow (P \leq R \wedge Q \leq R \wedge P \leq S \wedge Q \leq S) \\
&\quad \vee \left( \neg(P \leq R) \wedge \neg(Q \leq R) \wedge \neg(P \leq S) \wedge \neg(Q \leq S) \right)
\end{aligned} \tag{34}$$

【0 2 3 3】

上式(34)において、2行目は、データP、Qのうちのいずれか一方が最尤パスとなる場合を示しており、3行目は、データR、Sのうちのいずれか一方が最尤パスとなる場合を示している。また、上式(34)において、 $\min(\cdot, \cdot)$ は、小さい値を有するものを選択する関数であり、 $\max(\cdot, \cdot)$ は、大きい値を有するものを選択する関数である。すなわち、上式(34)において、4行目における $\max(P, Q)$ は、データPとデータQとの間で行われる1回戦の敗者、すなわち、値が大きいものを示しており、 $\min(P, Q)$ は、データPとデータQとの間で行われる1回戦の勝者、すなわち、値が小さいものを示しており、 $\max(R, S)$ は、データRとデータSとの間で行われる1回戦の敗者、すなわち、値が大きいものを示しており、 $\min(R, S)$ は、データRとデータSとの間で行われる1回戦の勝者、すなわち、値が小さいものを示している。

【0 2 3 4】

このような同値関係が成立することによって、絶対値データ選択部160は、

セクタ167による選択動作を実現する選択部として、図21に示すように構成することができる。すなわち、選択部180は、ANDゲート181と、NANDゲート182と、ORゲート183とを有するものとして実装することができる。

#### 【0235】

ANDゲート181は、比較回路133, 134, 135, 136から供給された4つの比較結果情報の論理積をとる。ANDゲート181は、求めた論理積をORゲート183に供給する。

#### 【0236】

NANDゲート182は、比較回路133, 134, 135, 136から供給された4つの比較結果情報の論理積の否定をとる。NANDゲート182は、求めた論理積の否定をORゲート183に供給する。

#### 【0237】

ORゲート183は、ANDゲート181から供給された論理積と、NANDゲート182から供給された論理積の否定との論理和をとる。ORゲート183は、求めた論理和を比較結果情報 $C_4$ としてセクタ170に供給する。

#### 【0238】

このような選択部180は、比較回路133, 134, 135, 136から供給された4つの比較結果情報のうち、いずれか1つの比較結果情報を選択し、この比較結果情報を比較結果情報 $C_4$ としてセクタ170に供給することができる。

#### 【0239】

以上のような絶対値データ選択部160による遅延量を見積もると、絶対値算出回路161, 162, 163, 164, 165, 166による遅延量は、上述した絶対値算出回路90による遅延量と等しいことから、1つの加算器と1つのセクタ分の遅延量を有することから、全体で1つの加算器と3つの加算器分の遅延量となる。すなわち、絶対値データ選択部160は、絶対値算出回路161, 162, 163, 164, 165, 166による遅延量の他に、2つのセクタ分の遅延量で済み、上述した絶対値データ選択部150と比較して、遅延量を

小さくすることができる。

#### 【0240】

さて、上述したパス選択部130及び絶対値データ選択部160を組み合わせると、加算比較選択回路120は、図22に示すように実装することができる。

#### 【0241】

すなわち、加算比較選択回路120は、同図に示すように、上述した加算比較選択回路60における加算器61, 62, 69, 70のそれぞれと等価である加算器121, 122, 123, 124と、パス選択部130と、絶対値データ選択部160と、線形近似手段である上述した線形近似回路100と、上述した加算比較選択回路60における差分器80に対応し、パス選択部130から供給されたデータと線形近似回路100から供給されたデータとの差分値を求める差分器125とを有する。

#### 【0242】

この加算比較選択回路120において、絶対値データ選択部160におけるセクタ169は、実際には、3つのセクタ191, 192, 193を有するものとなる。

#### 【0243】

すなわち、セクタ191は、比較回路131から供給された比較結果情報 $C_1$ に基づいて、絶対値算出回路163から供給される絶対値データ $|P-R|$ と、絶対値算出回路164から供給される絶対値データ $|Q-R|$ とのうち、いずれか一方の絶対値データを選択する。具体的には、セクタ191は、比較回路131による比較の結果、 $P \leq Q$ の場合には、絶対値算出回路163から供給される絶対値データ $|P-R|$ を選択し、 $P > Q$ の場合には、絶対値算出回路164から供給される絶対値データ $|Q-R|$ を選択する。セクタ191は、選択した絶対値データをセクタ193に供給する。

#### 【0244】

また、セクタ192は、比較回路131から供給された比較結果情報 $C_1$ に基づいて、絶対値算出回路165から供給される絶対値データ $|P-S|$ と、絶対値算出回路166から供給される絶対値データ $|Q-S|$ とのうち、いずれか

一方の絶対値データを選択する。具体的には、セレクタ192は、比較回路131による比較の結果、 $P \leq Q$ の場合には、絶対値算出回路165から供給される絶対値データ  $|P - S|$  を選択し、 $P > Q$ の場合には、絶対値算出回路166から供給される絶対値データ  $|Q - S|$  を選択する。セレクタ192は、選択した絶対値データをセレクタ193に供給する。

## 【0245】

さらに、セレクタ193は、比較回路132から供給された比較結果情報 $C_2$ に基づいて、セレクタ191から供給される絶対値データと、セレクタ192から供給される絶対値データとのうち、いずれか一方の絶対値データを選択する。具体的には、セレクタ193は、比較回路132による比較の結果、 $R \leq S$ の場合には、セレクタ191から供給される絶対値データを選択し、 $R > S$ の場合には、セレクタ192から供給される絶対値データを選択する。セレクタ193は、選択した絶対値データをセレクタ170に供給する。

## 【0246】

このような加算比較選択回路120における遅延量は、比較回路131, 132, 133, 134, 135, 136による処理と、絶対値算出回路161, 162, 163, 164, 165, 166による処理とが並列的に行われるとともに、セレクタ137, 139, 140, 191, 192による処理が並列的に行われるとともに、セレクタ138, 141, 193による処理が並列的に行われることから、図12に示した線形近似回路100を用いた場合には、4つの加算器と6つのセレクタ分となり、図13に示した線形近似回路100を用いた場合には、3つの加算器と6つのセレクタ分となる。したがって、加算比較選択回路120は、上述した加算比較選択回路60と比較して、遅延量を小さくすることができる。

## 【0247】

以上説明したように、符号化装置1と復号装置3とを用いて構成されるデータ送受信システムは、復号装置3において、最尤パスの選択と補正項の算出に用いる絶対値データの算出とを、入力された複数のデータによる勝ち抜き戦の原理を利用して行うことによって、迅速な処理が可能となり、高速化を図ることができ

る。特に、復号装置 3 は、線形近似による  $\log - \text{sum}$  補正を行う際に、最尤パスと準最尤パスとの間だけで補正を行うことによって、性能を劣化させることなく、動作速度を向上させることができる。

## 【 0 2 4 8 】

すなわち、これらの符号化装置 1 と復号装置 3 とを用いて構成されるデータ送受信システムは、高性能且つ高速に畳み込み符号の復号を実現するものであり、ユーザに高い信頼性及び利便性を提供することができるものである。

## 【 0 2 4 9 】

なお、本発明は、上述した実施の形態に限定されるものではなく、例えば、符号化装置としては、畳み込み演算を行うものでなくてもよく、入力データが 2 ビット以上、すなわち、各ステートに少なくとも 3 つ以上のパスが到達するような符号化を行うものであればよい。

## 【 0 2 5 0 】

また、上述した実施の形態では、各ステートに 4 つのパスが到達し、これらの 4 つのパスのうち、最尤パスと準最尤パスとを用いて補正を行うものとして説明したが、本発明は、各ステートに到達したパスのうち、尤度の高い少なくとも 2 つ以上のパスを用いて補正を行うものであれば適用可能である。

## 【 0 2 5 1 】

さらに、上述した実施の形態では、 $\text{Log} - \text{BCJR}$  アルゴリズムに基づく MAP 復号を行い、いわゆる線形近似による  $\log - \text{sum}$  補正を行うことで対数軟出力を求めるものとして説明したが、本発明は、 $\text{Max} - \text{Log} - \text{BCJR}$  アルゴリズムに基づく MAP 復号を行うものにも適用可能であり、高速化を図ることができる。この場合、加算比較選択回路は、上述した絶対値データ選択部や線形近似回路を有する必要はなく、最尤パスを求める構成とすればよいことは勿論である。

## 【 0 2 5 2 】

さらにまた、本発明は、いわゆる並列接続畳み込み符号、縦列接続畳み込み符号、ターボ符号化変調方式による符号又は縦列接続符号化変調方式による符号といったように、複数の要素符号を接続して構成される符号の復号を行う場合にも



容易に適用できるものである。

【0253】

また、上述した実施の形態では、符号化装置及び復号装置をデータ送受信システムにおける送信装置及び受信装置に適用して説明したが、本発明は、例えばフロッピーディスク、CD-ROM又はMO (Magneto Optical) といった磁気、光又は光磁気ディスク等の記録媒体に対する記録及び／又は再生を行う記録及び／又は再生装置に適用することもできる。この場合、符号化装置により符号化されたデータは、無記憶通信路に等価とされる記録媒体に記録され、復号装置により復号されて再生される。

【0254】

以上のように、本発明は、その趣旨を逸脱しない範囲で適宜変更が可能であることはいうまでもない。

【0255】

【発明の効果】

以上詳細に説明したように、本発明にかかる復号装置は、各ステートに少なくとも3つ以上のパスが到達する符号化がなされて受信した軟入力とされる受信値に基づいて任意のステートを通過する確率を対数表記した対数尤度を求め、この対数尤度を用いて復号を行う復号装置であって、各ステートに到達した少なくとも3つ以上のパスのうち、尤度の高い少なくとも2つ以上のパスを求め、これらの少なくとも2つ以上のパスの中から、最も尤度の高いパスである最尤パスを選択するパス選択手段を備える。

【0256】

したがって、本発明にかかる復号装置は、パス選択手段によって、尤度の高い少なくとも2つ以上のパスを求め、最尤パスを選択することによって、性能を劣化させることなく、高速化を図ることができる。

【0257】

また、本発明にかかる復号方法は、各ステートに少なくとも3つ以上のパスが到達する符号化がなされて受信した軟入力とされる受信値に基づいて任意のステートを通過する確率を対数表記した対数尤度を求め、この対数尤度を用いて復号

を行う復号方法であって、各ステートに到達した少なくとも3つ以上のパスのうち、尤度の高い少なくとも2つ以上のパスを求め、これらの少なくとも2つ以上のパスの中から、最も尤度の高いパスである最尤パスを選択するパス選択工程を備える。

【0258】

したがって、本発明にかかる復号方法は、パス選択工程にて、尤度の高い少なくとも2つ以上のパスを求め、最尤パスを選択することによって、性能を劣化させることなく、高速化を図ることを可能とする。

【図面の簡単な説明】

【図1】

本発明の実施の形態として示すデータ送受信システムを適用する通信モデルの構成を説明するブロック図である。

【図2】

同データ送受信システムにおける符号化装置の構成を説明するブロック図である。

【図3】

符号化装置におけるトレリスを説明する図である。

【図4】

同データ送受信システムにおける復号装置の構成を説明するブロック図である。

【図5】

復号装置が備える  $I\alpha$  算出・記憶回路の構成を説明するブロック図である。

【図6】

$I\alpha$  算出・記憶回路が有する  $I\alpha$  算出回路の構成を説明するブロック図である。

【図7】

復号装置が備える  $I\beta$  算出・記憶回路の構成を説明するブロック図である。

【図8】

$I\beta$  算出・記憶回路が有する  $I\beta$  算出回路の構成を説明するブロック図である。

## 【図 9】

I  $\alpha$  算出回路又は I  $\beta$  算出回路が有する加算比較選択回路の構成を説明するブロック図であって、Log-BCJR アルゴリズムに基づいて実直に実装した加算比較選択回路の構成を説明するブロック図である。

## 【図 10】

加算比較選択回路が有する絶対値算出回路の構成を説明するブロック図である。

## 【図 11】

加算比較選択回路が有する線形近似回路による log-sum 補正を説明する図であって、補正項を示す関数と線形近似した関数とを示すグラフである。

## 【図 12】

線形近似回路の構成を説明するブロック図であって、関数  $F = -a |PP - QQ| + b$  の係数  $-a$  を 2 のべき乗を用いて表現して補正項の値を算出する線形近似回路の構成を説明するブロック図である。

## 【図 13】

図 12 に示す線形近似回路とは異なる他の線形近似回路の構成を説明するブロック図であって、関数  $F = -a |PP - QQ| + b$  の係数  $-a$  と係数  $b$  とを 2 のべき乗を用いて表現して補正項の値を算出する線形近似回路の構成を説明するブロック図である。

## 【図 14】

図 13 に示す線形近似回路による演算を説明する図である。

## 【図 15】

加算比較選択回路が有するパス選択部の構成を説明するブロック図である。

## 【図 16】

パス選択部における動作を説明するための図であって、入力された複数のデータの大小を比較して選択する動作を説明する図である。

## 【図 17】

パス選択部における動作を説明するための図であって、パス選択部による選択

結果の一例を示す図である。

【図 1 8】

加算比較選択回路が有する絶対値データ選択部の構成を説明するブロック図であって、2つの絶対値データを実直に比較して選択する絶対値データ選択部の構成を説明するブロック図である。

【図 1 9】

図 1 8 に示す絶対値データ選択部とは異なる絶対値データ選択部の構成を説明するブロック図である。

【図 2 0】

絶対値データ選択部における動作を説明するための図であって、パス選択部による8つの選択結果を示す図である。

【図 2 1】

図 1 9 に示す絶対値データ選択部が有する選択部の構成を説明するブロック図である。

【図 2 2】

図 9 に示す加算比較選択回路とは異なる加算比較選択回路の構成を説明するブロック図であって、図 1 5 に示すパス選択部と図 1 9 に示す絶対値データ選択部を組み合わせ構成した加算比較選択回路の構成を説明するブロック図である。

【図 2 3】

通信モデルの構成を説明するブロック図である。

【図 2 4】

従来の符号化装置におけるトレリスを説明する図であって、確率  $\alpha_t$ 、 $\beta_t$  及び  $\gamma_t$  の内容を説明するための図である。

【図 2 5】

従来の復号装置において、BCJR アルゴリズムを適用して軟出力復号を行う際の一連の工程を説明するフローチャートである。

【図 2 6】

従来の復号装置において、Max-Log-BCJR アルゴリズムを適用して軟出力復号を行う際の一連の工程を説明するフローチャートである。

## 【図 27】

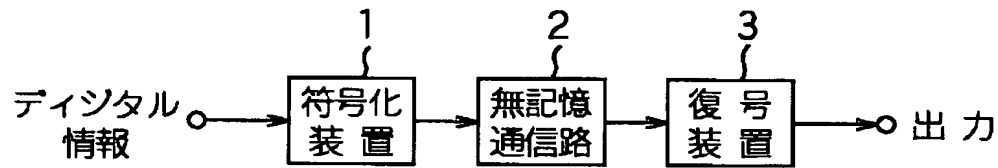
log-sum補正を説明する図であって、(A)は、補正項を示す関数と線形近似法により近似した関数とを示すグラフであり、(B)は、補正項を示す関数と閾値近似法により近似した関数とを示すグラフである。

## 【符号の説明】

1 符号化装置、 3 復号装置、 32  $I_r$  算出・記憶回路、 33  $I_\alpha$  算出・記憶回路、 34  $I_\beta$  算出・記憶回路、 35 軟出力算出回路、  
 43  $I_\alpha$  算出回路、 47<sub>0</sub>, 47<sub>1</sub>, 47<sub>2</sub>, 47<sub>3</sub>, 55<sub>0</sub>, 55<sub>1</sub>, 55<sub>2</sub>,  
 55<sub>3</sub>, 60, 120 加算比較選択回路、 51<sub>1</sub>, 51<sub>2</sub>  $I_\beta$  算出回路、  
 65, 73, 79 補正項算出回路、 67, 75, 81, 90 絶対値算出回路、  
 68, 76, 82, 100 線形近似回路、 130 パス選択部、 150, 160 絶対値データ選択部

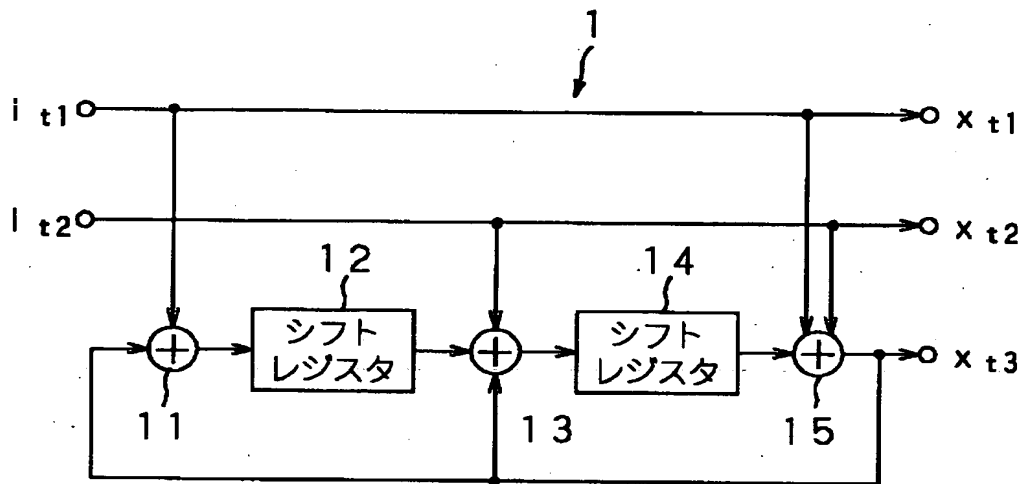
【書類名】 図面

【図 1】



通信モデルの構成ブロック図

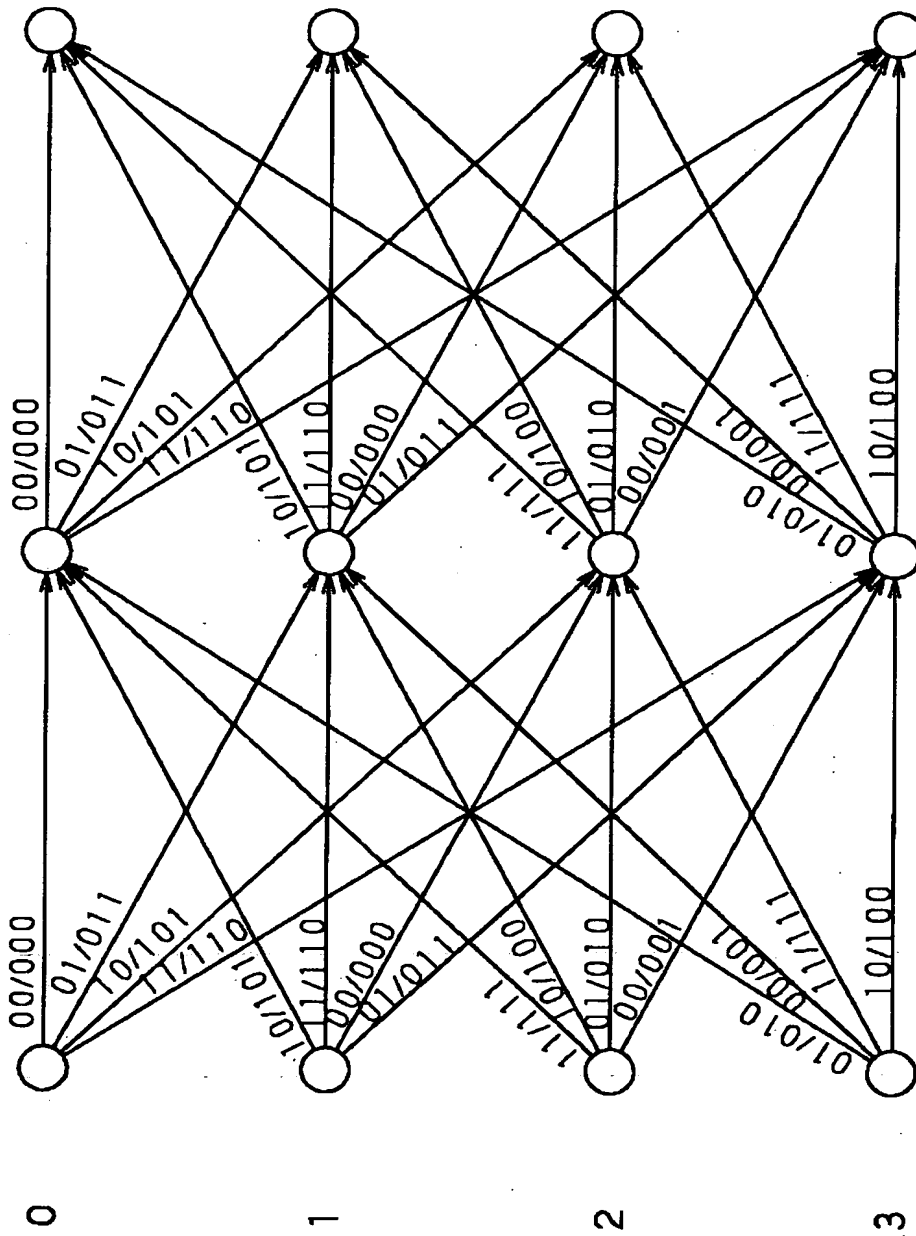
【図 2】



符号化装置の構成ブロック図

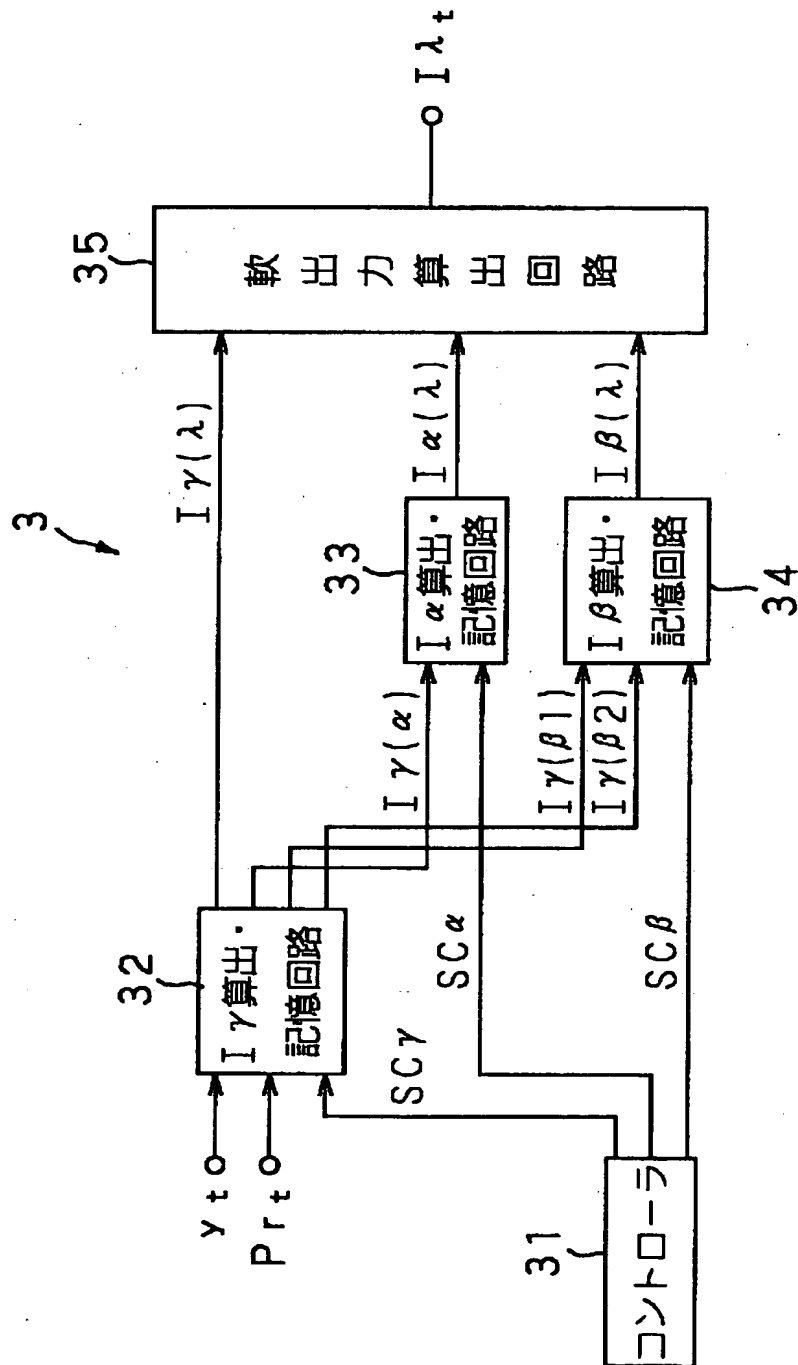
【図3】

スタート番号



トレリスの説明図

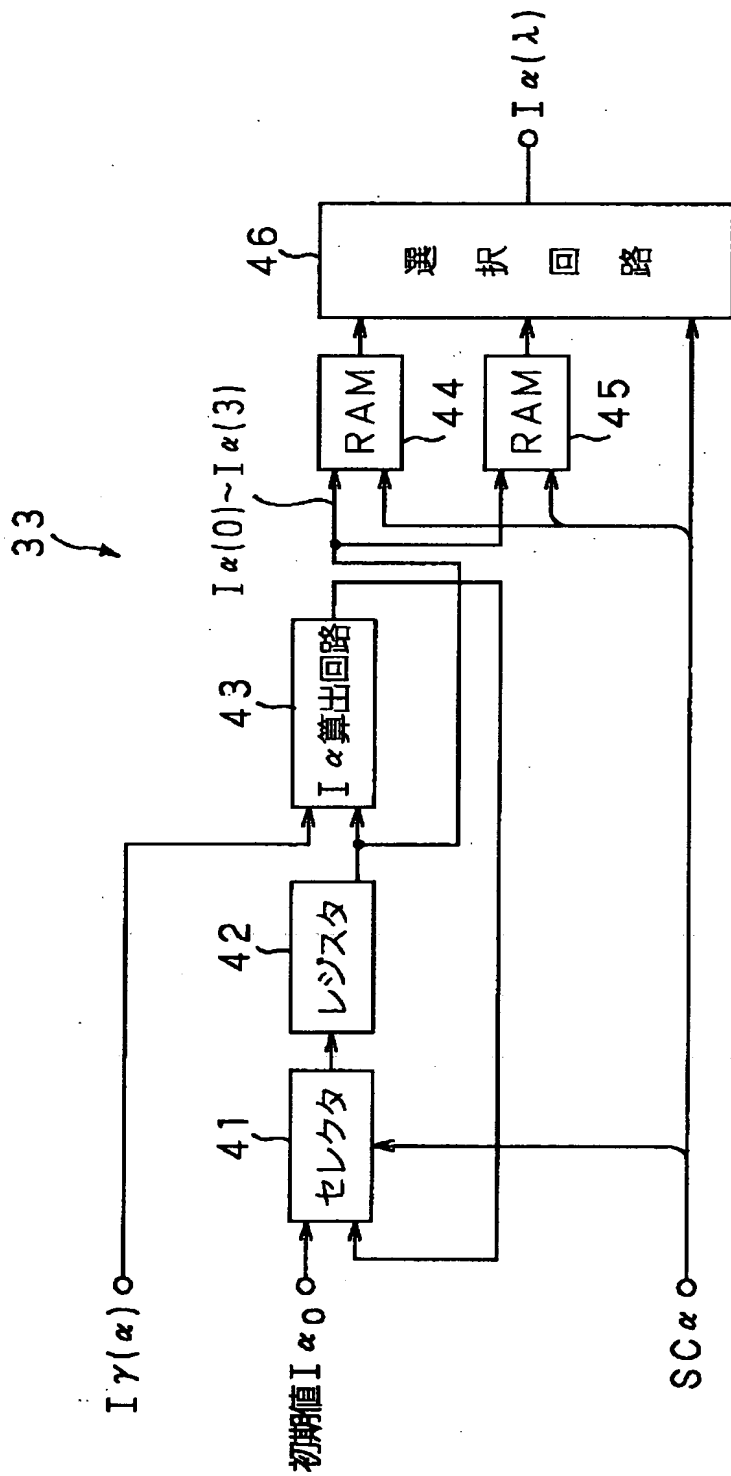
【図4】



復号装置の構成ブロック図

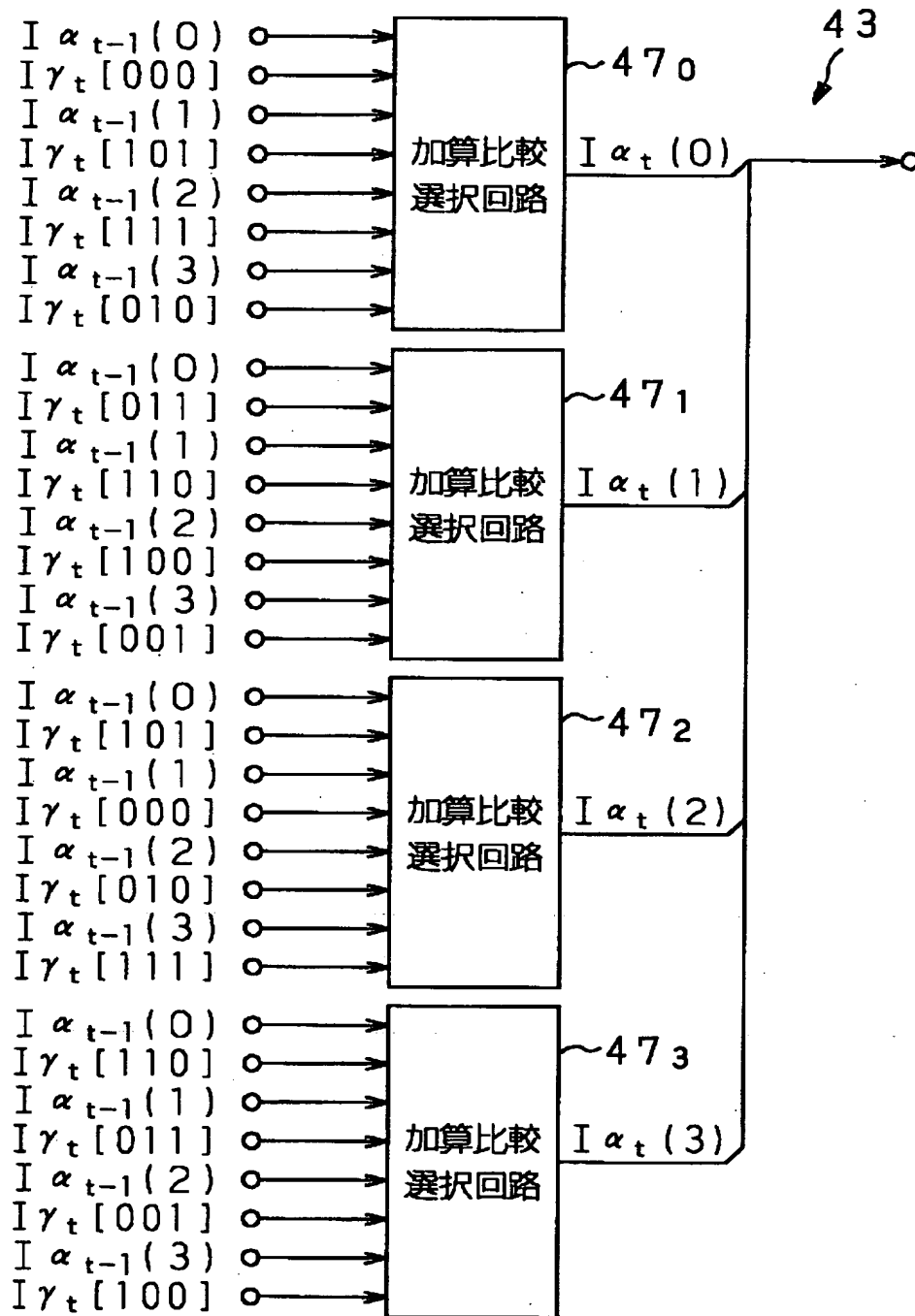


【図5】



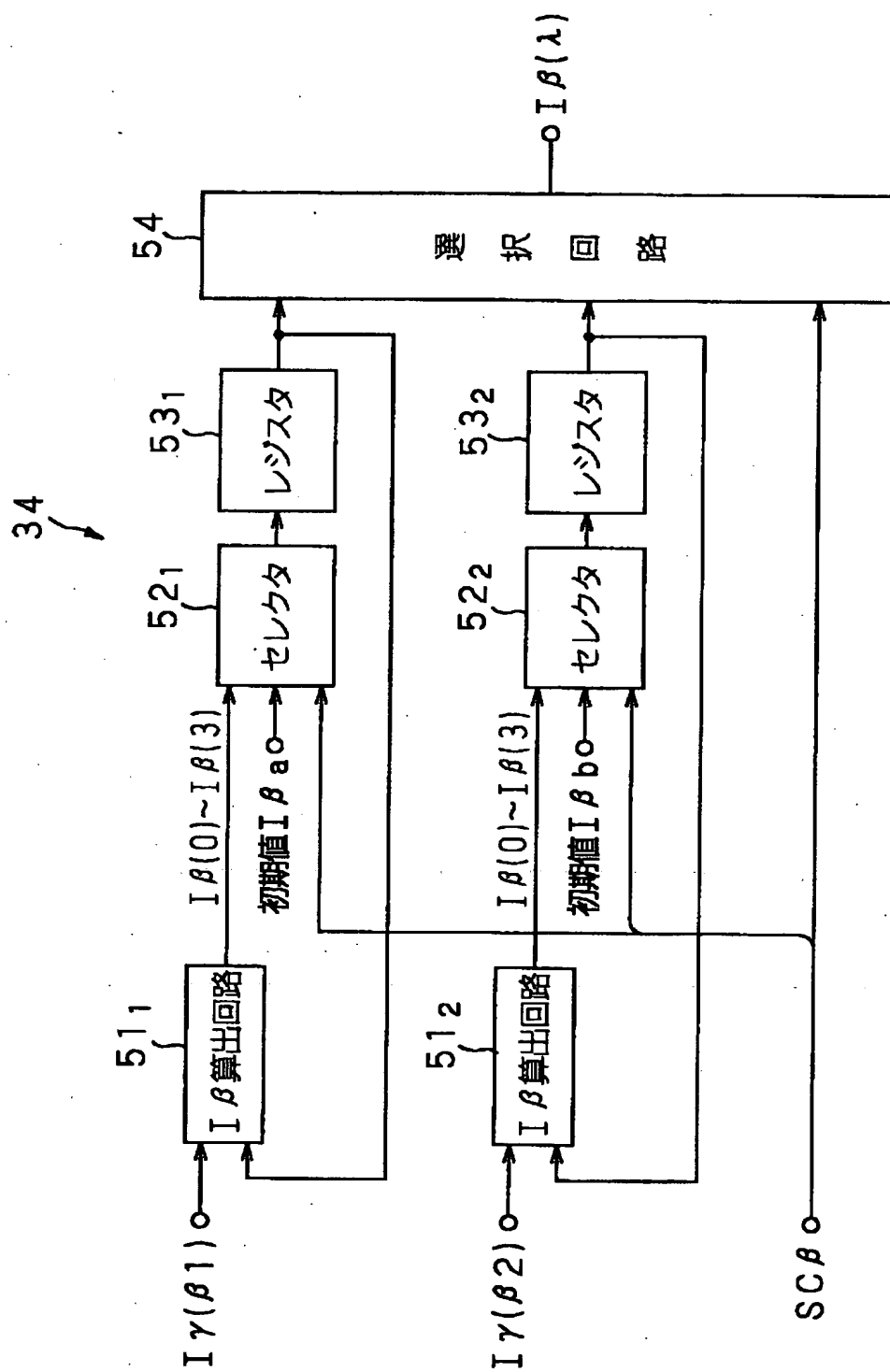
$I\alpha$ 算出・記憶回路の構成ブロック図

【図6】



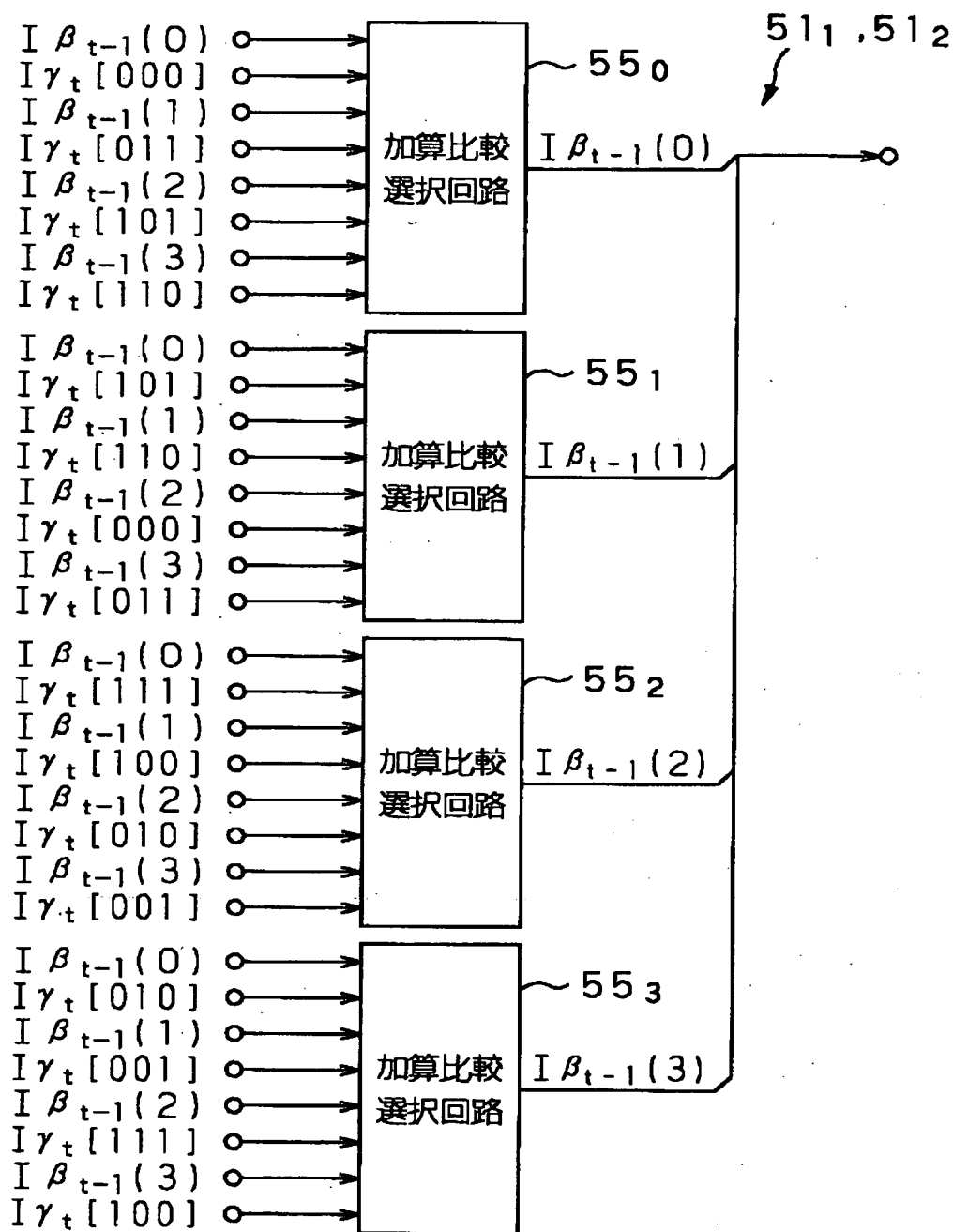
$I\alpha$ 算出回路の構成ブロック図

【図 7】



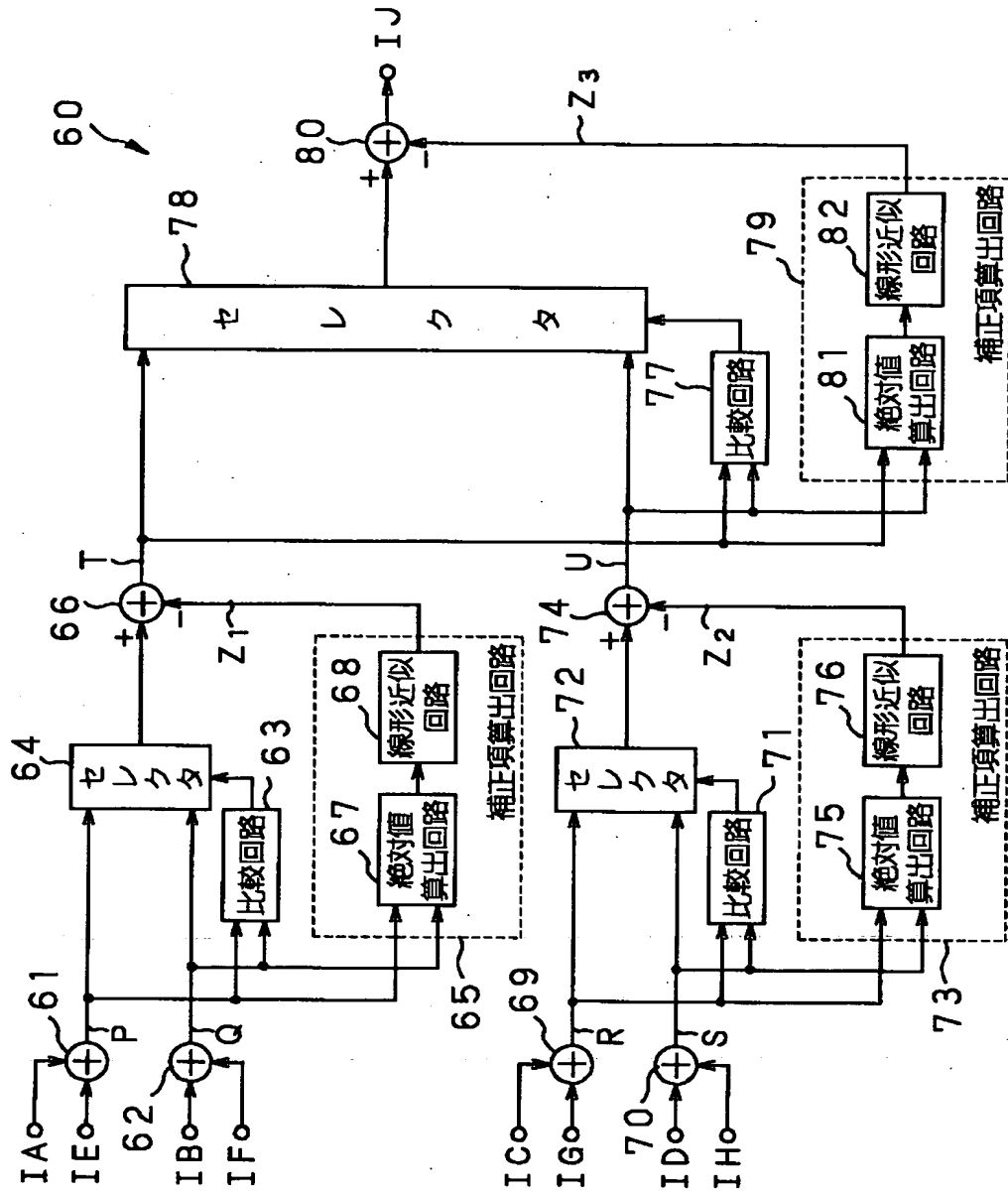
Ⅰβ算出・記憶回路の構成ブロック図

【図 8】



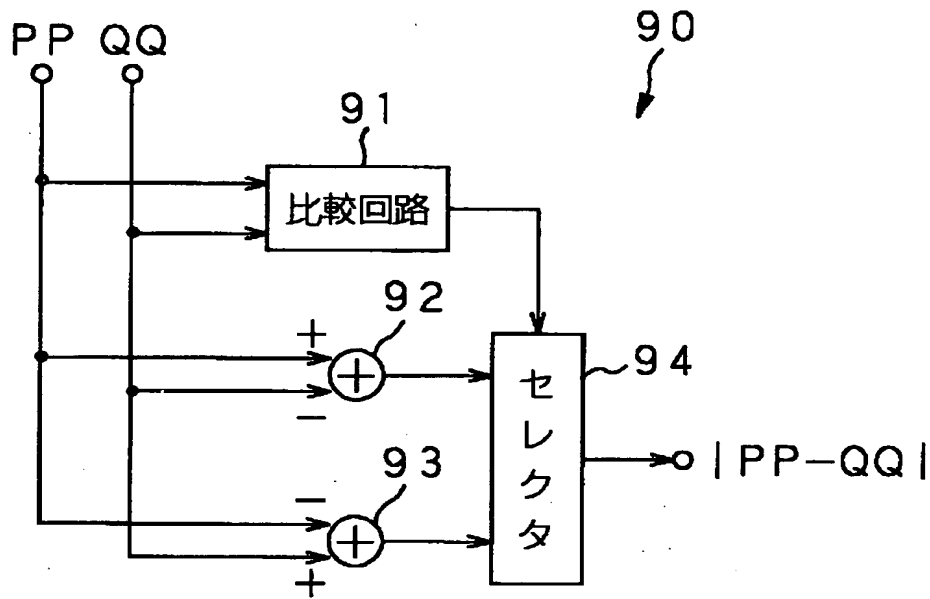
$I\beta$  算出回路の構成ブロック図

【図9】



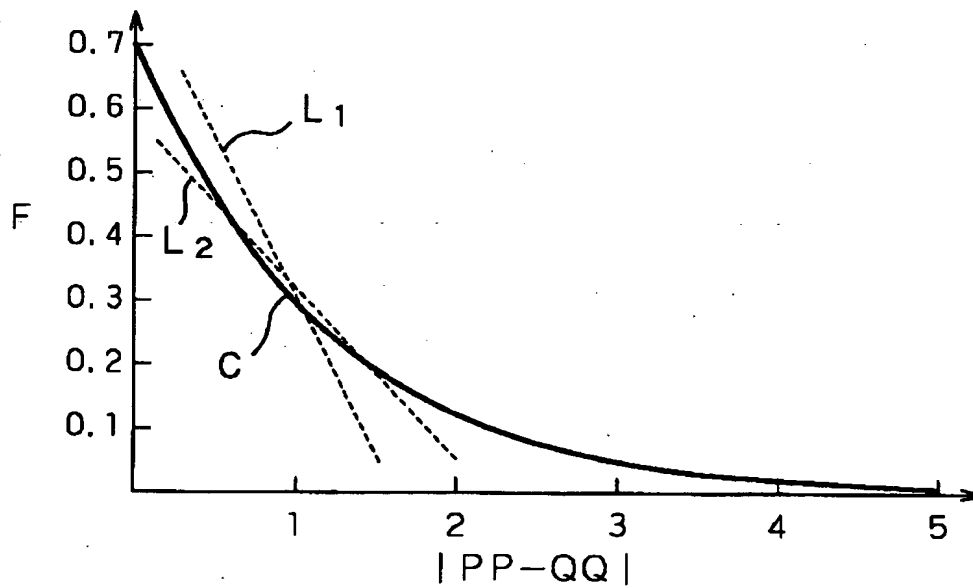
加算比較選択回路の構成ブロック図

【図10】

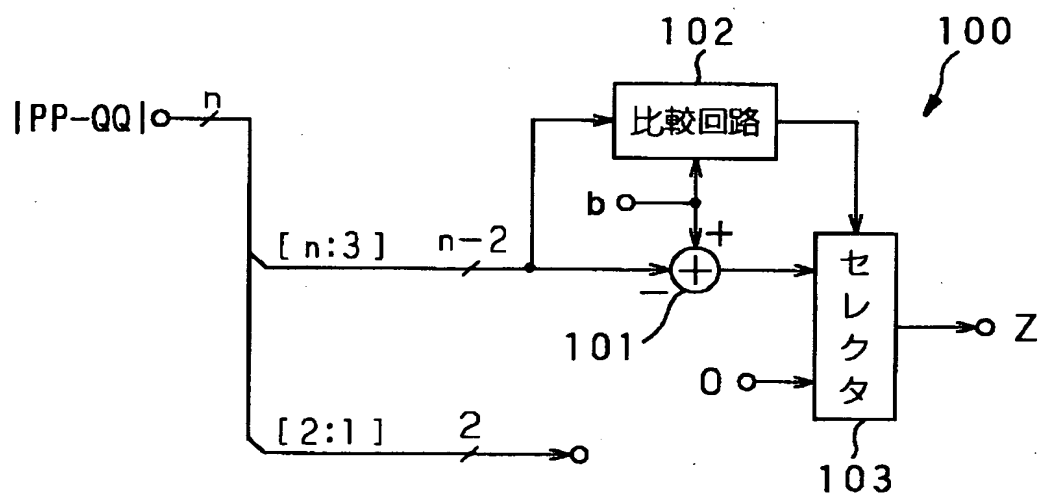


絶対値算出回路の構成ブロック図

【図11】

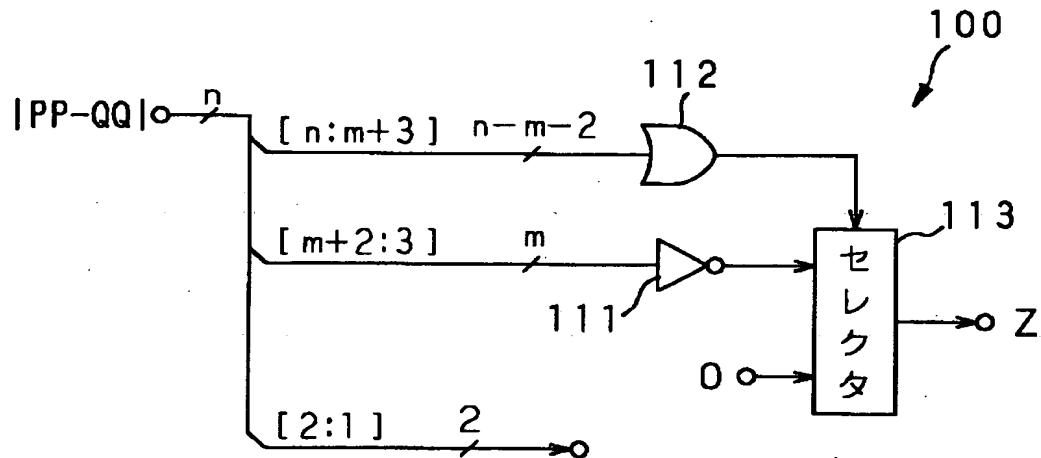


【図 12】



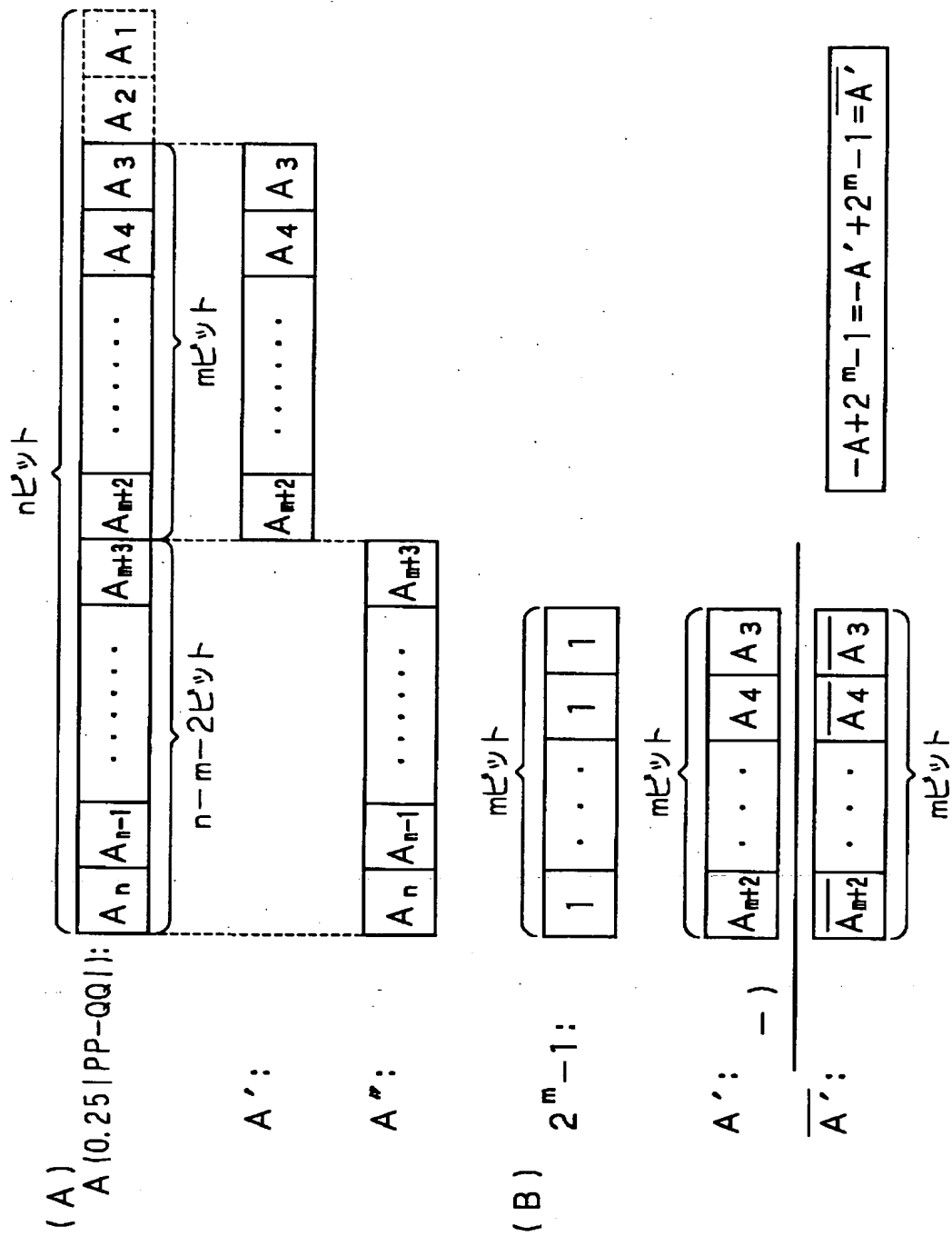
線形近似回路の構成ブロック図

【図 13】



線形近似回路の構成ブロック図

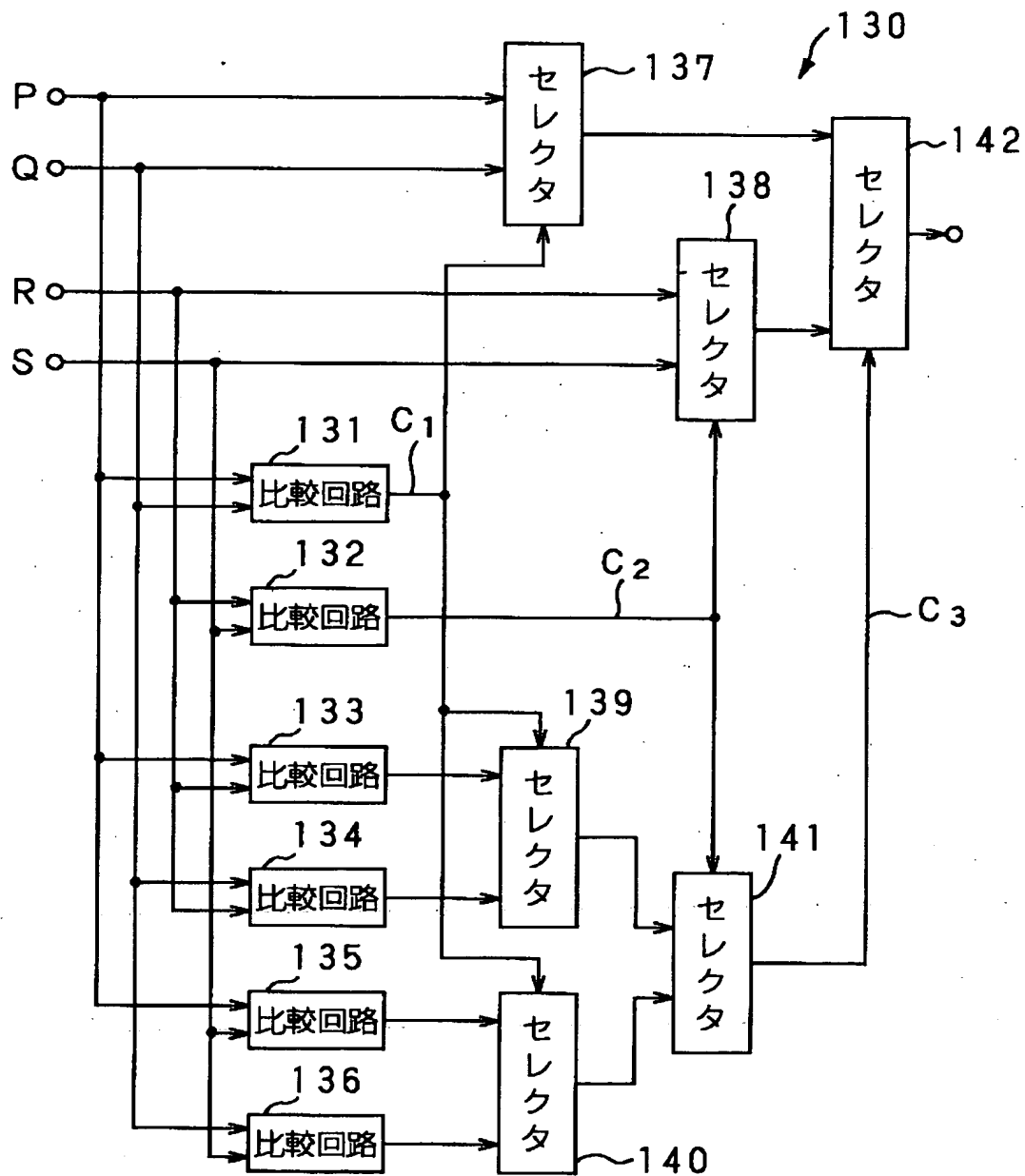
【図 14】



線形近似回路による演算の説明図



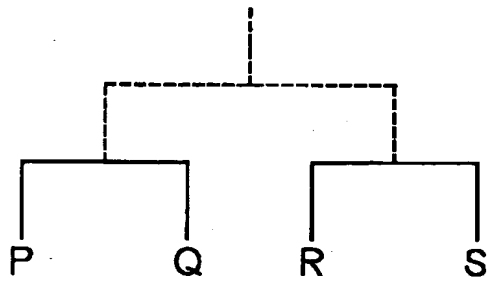
【図15】



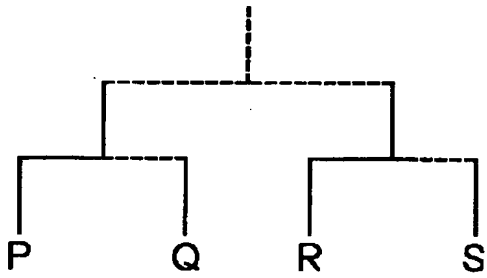
バス選択部の構成ブロック図

【図16】

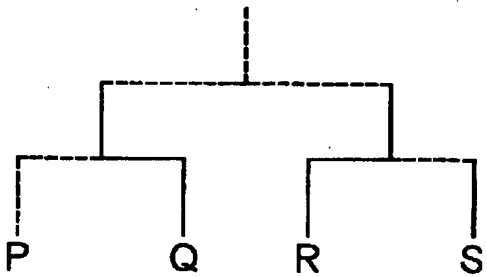
(A)



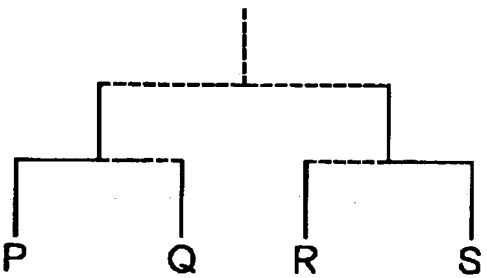
(B)



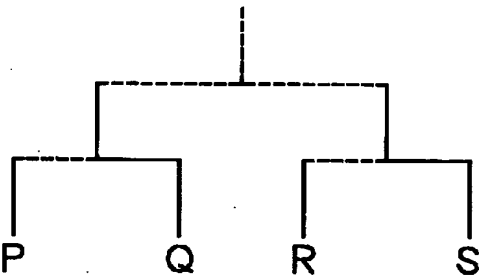
(C)



(D)

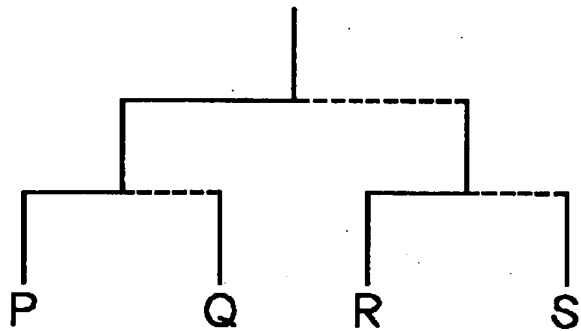


(E)



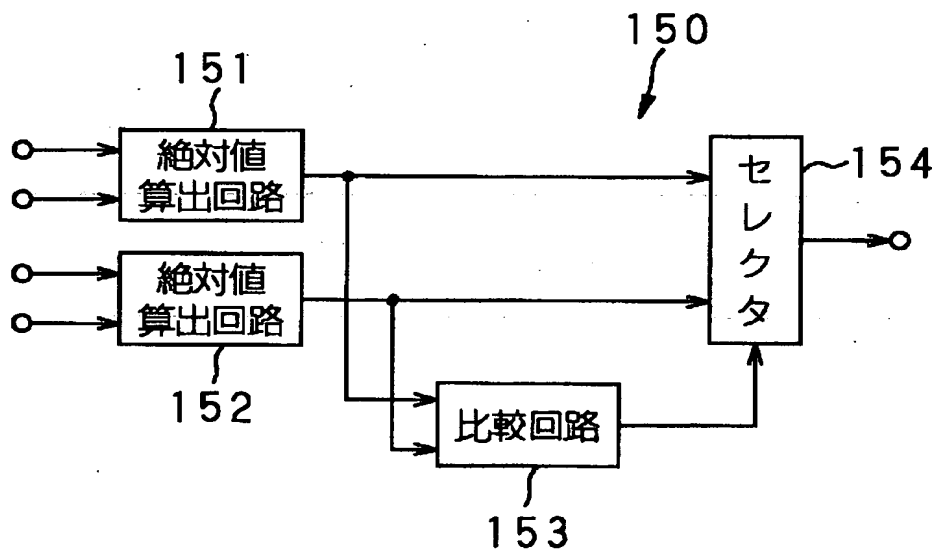
パス選択部における動作の説明図

【図 17】



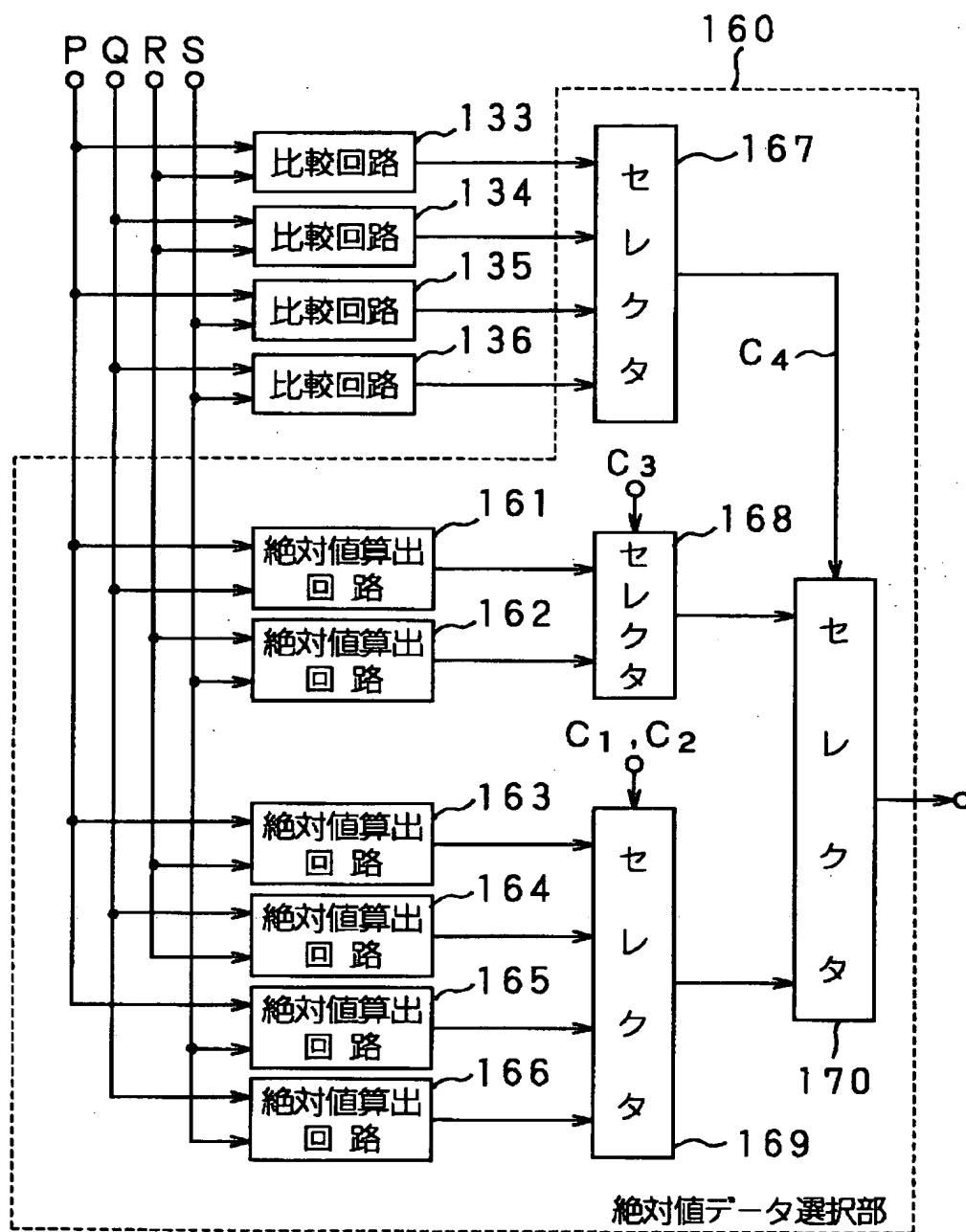
パス選択部における動作の説明図

【図 18】



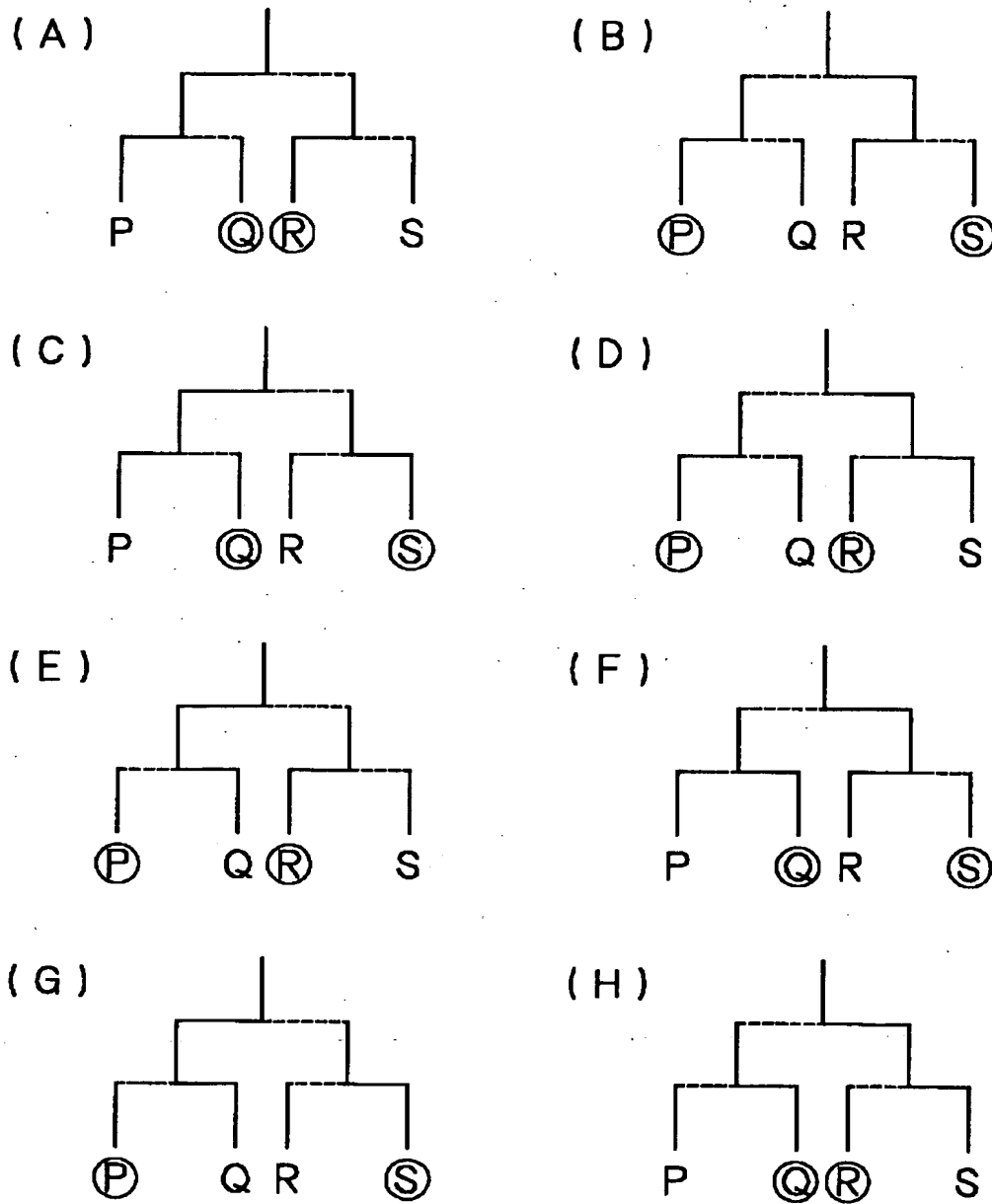
絶対値データ選択部の構成ブロック図

【図 1 9】



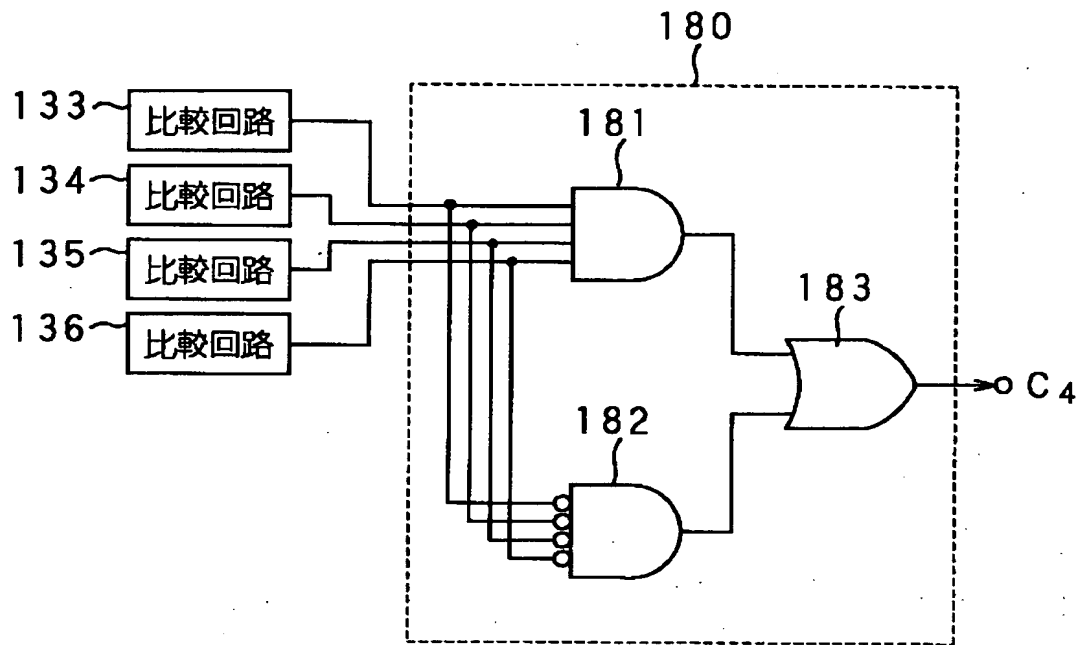
絶対値データ選択部の構成ブロック図

【図20】



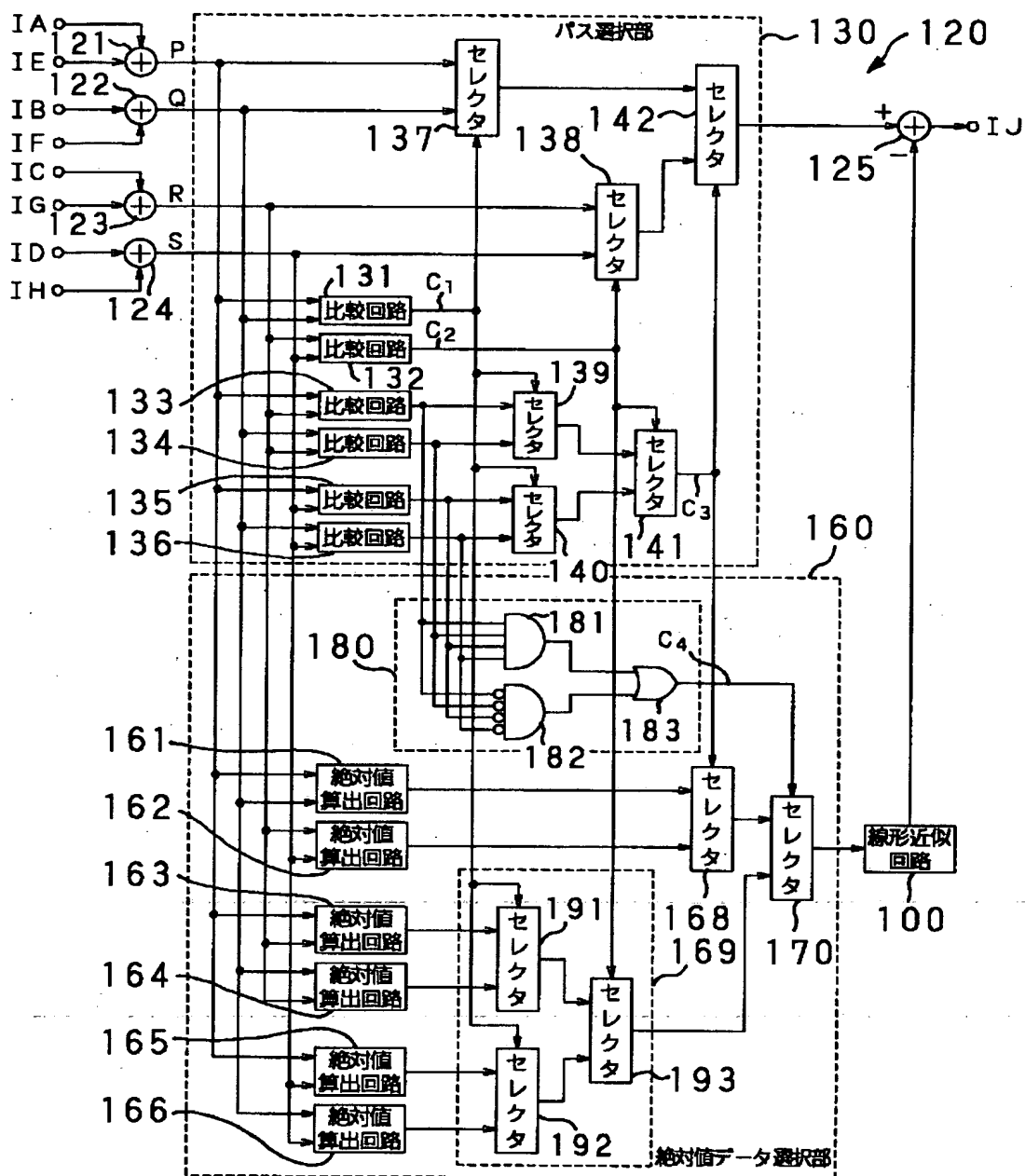
絶対値データ選択部における動作の説明図

【図 2 1】



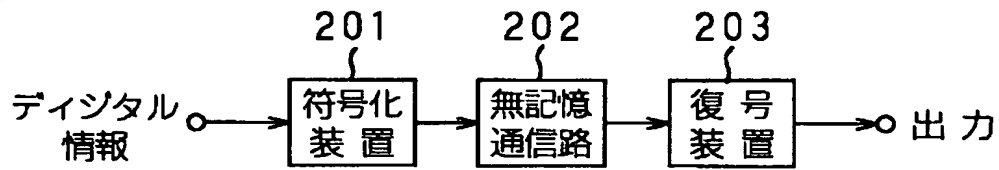
選択部の構成ブロック図

【図 22】



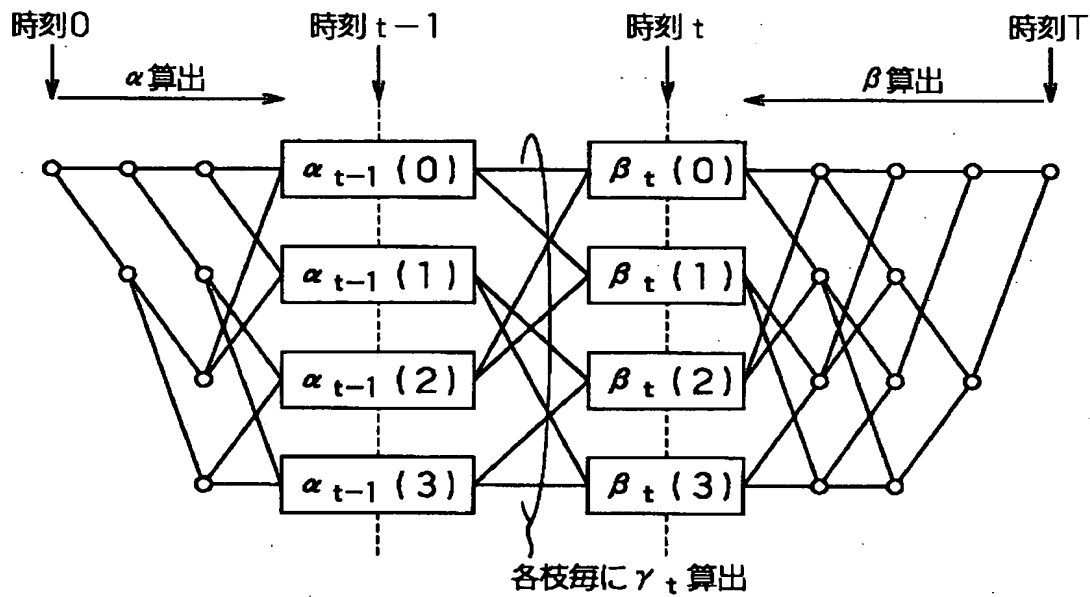
加算比較選択回路の構成ブロック図

【図 23】



通信モデルの構成ブロック図

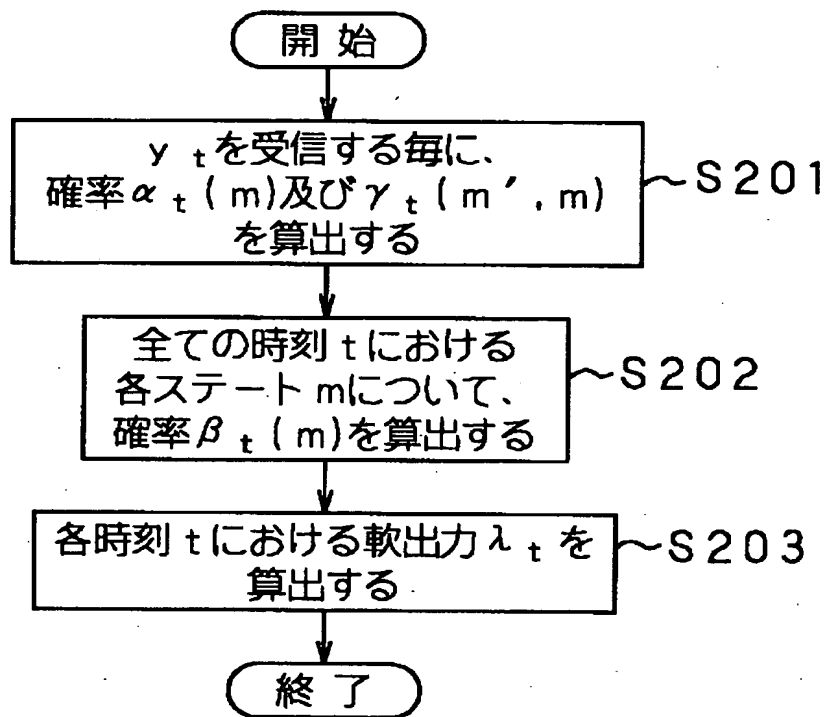
【図 24】



トレリスの説明図

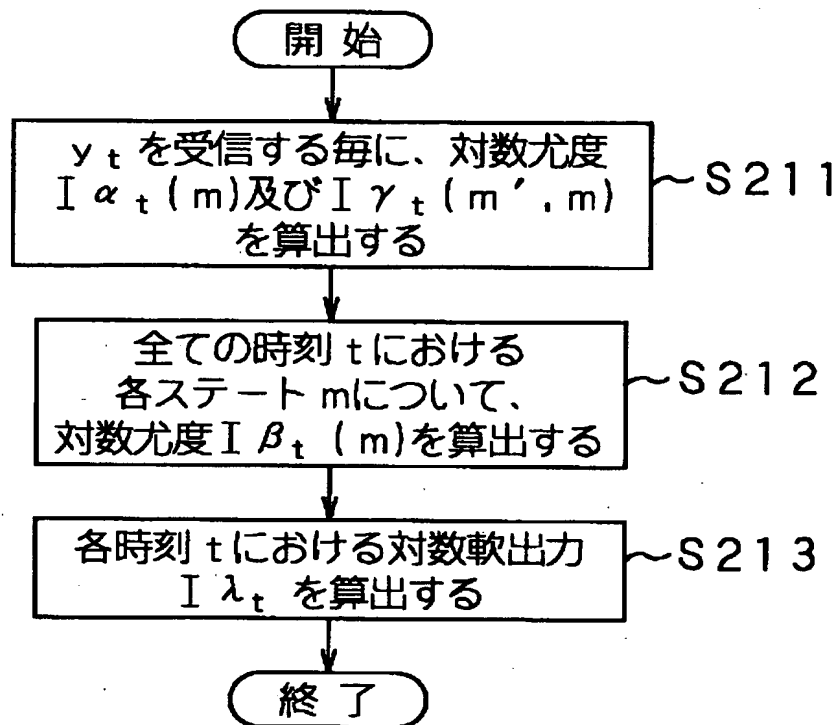


【図 25】



復号装置における一連の処理工程

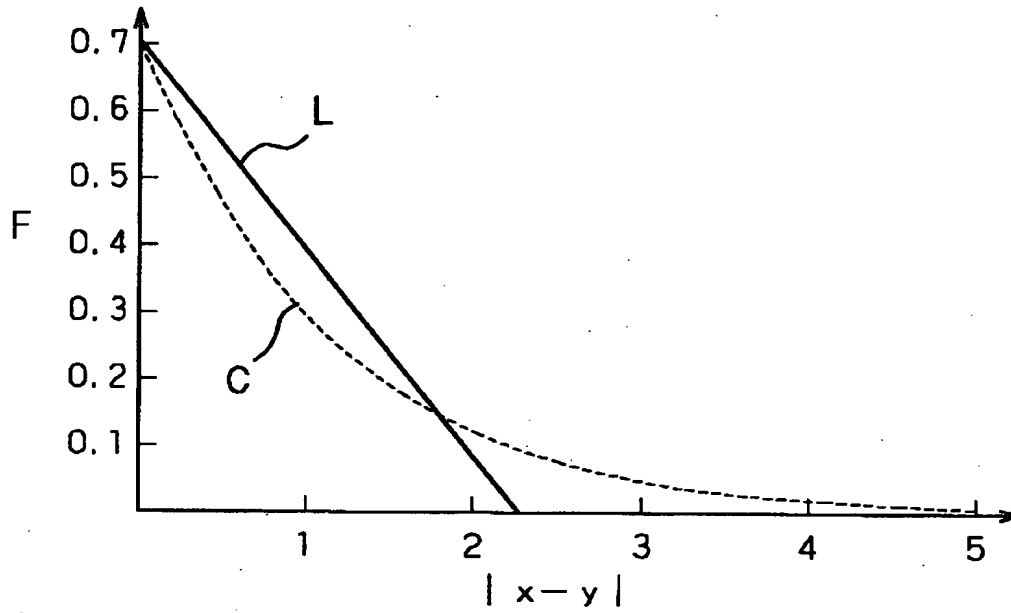
【図 26】



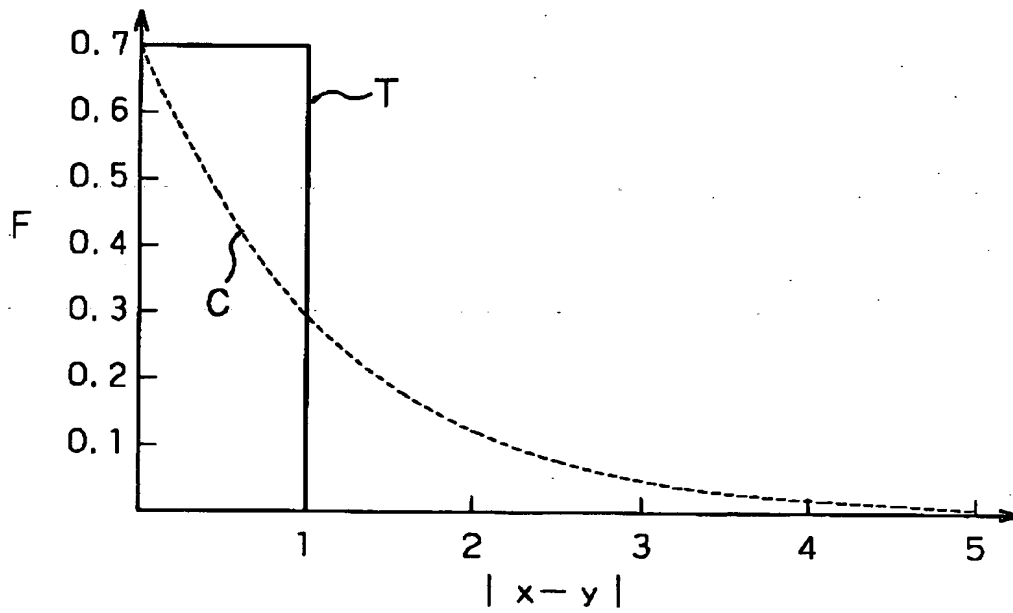
復号装置における一連の処理工程

【図 27】

(A)



(B)



log-sum補正の説明図

【書類名】 要約書

【要約】

【課題】 復号の性能を劣化させることなく、高速化を図る。

【解決手段】 復号装置は、各ステートに到達した少なくとも3つ以上のパスのうち、尤度の高い少なくとも2つ以上のパスを求め、これらの少なくとも2つ以上のパスの中から最尤パスを選択するパス選択部130と、最尤パスに対応するデータと準最尤パスに対応するデータとの差分値の絶対値を求めて選択する絶対値データ選択部160とを備える。パス選択部130は、データP、Q、R、Sについて、いわば勝ち抜き戦に喩えられる動作を行うことによって、データP、Q、R、Sの値の大小を比較して最尤パスに対応するデータを選択する。また、絶対値データ選択部160は、パス選択部130における比較回路133乃至136により求められた比較結果情報に基づいて、絶対値算出回路161乃至166により算出された絶対値データの大小関係を判別する。

【選択図】 図22

出 願 人 履 歴 情 報

識別番号 [000002185]

1. 変更年月日 1990年 8月30日  
[変更理由] 新規登録  
住 所 東京都品川区北品川6丁目7番35号  
氏 名 ソニー株式会社